Fast Models

Version 11.13

Fixed Virtual Platforms (FVP) Reference Guide

arm

100966_1113_00_en

Fast Models

Fixed Virtual Platforms (FVP) Reference Guide

Release Information

Document History

Issue	Date	Confidentiality	Change
A	31 May 2014	Non-Confidential	New document for Fast Models v9.0, from DUI0575H for v8.3.
B	30 November 2014	Non-Confidential	Update for v9.1.
C	28 February 2015	Non-Confidential	Update for v9.2.
D	31 May 2015	Non-Confidential	Update for v9.3.
E	31 August 2015	Non-Confidential	Update for v9.4.
F	30 November 2015	Non-Confidential	Update for v9.5.
G	29 February 2016	Non-Confidential	Update for v9.6.
H	31 May 2016	Non-Confidential	Update for v10.0.
I	31 August 2016	Non-Confidential	Update for v10.1.
J	11 November 2016	Non-Confidential	Update for v10.2.
K	17 February 2017	Non-Confidential	Update for v10.3.
1100-00	31 May 2017	Non-Confidential	Update for v11.0. Document numbering scheme has changed.
1101-00	31 August 2017	Non-Confidential	Update for v11.1.
1102-00	17 November 2017	Non-Confidential	Update for v11.2.
1103-00	23 February 2018	Non-Confidential	Update for v11.3.
1104-00	22 June 2018	Non-Confidential	Update for v11.4.
1104-01	17 August 2018	Non-Confidential	Update for v11.4.2.
1105-00	23 November 2018	Non-Confidential	Update for v11.5.
1106-00	26 February 2019	Non-Confidential	Update for v11.6.
1107-00	17 May 2019	Non-Confidential	Update for v11.7.
1108-00	05 September 2019	Non-Confidential	Update for v11.8.
1108-01	03 October 2019	Non-Confidential	Update for v11.8.1.
1109-00	28 November 2019	Non-Confidential	Update for v11.9.
1110-00	12 March 2020	Non-Confidential	Update for v11.10.
1111-00	09 June 2020	Non-Confidential	Update for v11.11.
1112-00	22 September 2020	Non-Confidential	Update for v11.12.
1113-00	09 December 2020	Non-Confidential	Update for v11.13.

(LES-PRE-20349)

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

The information in this document is Final, that is for a developed product.

Web Address

developer.arm.com

Progressive terminology commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used terms that can be offensive. Arm strives to lead the industry and create change.

This document includes terms that can be offensive. We will replace these terms in a future issue of this document.

If you find offensive terms in this document, please contact *terms@arm.com*.

Contents

Fast Models Fixed Virtual Platforms (FVP) Reference Guide

Preface

About this book 9

Chapter 1 Introduction

1.1 About FVPs 1-12

1.2 Data collection in FVPs 1-13

Chapter 2 Getting Started with Fixed Virtual Platforms

2.1 Contents of the package 2-16

2.2 FVP command-line options 2-17

2.3 Loading and running an application on an FVP 2-22

2.4 Configuring the model 2-23

2.5 FVP debug 2-24

2.6 Using the CLCD window 2-25

2.7 Ethernet with VE FVPs 2-28

2.8 Using a terminal with a system model 2-30

2.9 Virtio P9 device component 2-33

Chapter 3 Base Platform FVPs

3.1 FVP_Base_AEMv8A-AEMv8A 3-36

3.2 FVP_Base_Cortex-A32x1 3-47

3.3 FVP_Base_Cortex-A32x2 3-55

3.4 FVP_Base_Cortex-A32x4 3-63
3.5 FVP_Base_Cortex-A35x1 3-72
3.6 FVP_Base_Cortex-A35x2 3-80
3.7 FVP_Base_Cortex-A35x4 3-88
3.8 FVP_Base_Cortex-A53x1 3-97
3.9 FVP_Base_Cortex-A53x2 3-105
3.10 FVP_Base_Cortex-A53x4 3-113
3.11 FVP_Base_Cortex-A55 3-122
3.12 FVP_Base_Cortex-A55+Cortex-A76 3-131
3.13 FVP_Base_Cortex-A55x1 3-140
3.14 FVP_Base_Cortex-A55x1+Cortex-A75x1 3-148
3.15 FVP_Base_Cortex-A55x2 3-157
3.16 FVP_Base_Cortex-A55x2+Cortex-A75x2 3-165
3.17 FVP_Base_Cortex-A55x4 3-174
3.18 FVP_Base_Cortex-A55x4+Cortex-A75x1 3-183
3.19 FVP_Base_Cortex-A55x4+Cortex-A75x2 3-192
3.20 FVP_Base_Cortex-A55x4+Cortex-A75x4 3-202
3.21 FVP_Base_Cortex-A55x4+Cortex-A76x2 3-212
3.22 FVP_Base_Cortex-A57x1 3-222
3.23 FVP_Base_Cortex-A57x1-A35x1 3-230
3.24 FVP_Base_Cortex-A57x1-A53x1 3-240
3.25 FVP_Base_Cortex-A57x2 3-250
3.26 FVP_Base_Cortex-A57x2-A35x4 3-258
3.27 FVP_Base_Cortex-A57x2-A53x4 3-268
3.28 FVP_Base_Cortex-A57x4 3-278
3.29 FVP_Base_Cortex-A57x4-A35x4 3-287
3.30 FVP_Base_Cortex-A57x4-A53x4 3-298
3.31 FVP_Base_Cortex-A65AEx2 3-309
3.32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 3-318
3.33 FVP_Base_Cortex-A65AEx4 3-328
3.34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 3-338
3.35 FVP_Base_Cortex-A65AEx8 3-348
3.36 FVP_Base_Cortex-A65x1 3-359
3.37 FVP_Base_Cortex-A65x2 3-367
3.38 FVP_Base_Cortex-A65x4 3-376
3.39 FVP_Base_Cortex-A72x1 3-385
3.40 FVP_Base_Cortex-A72x1-A53x1 3-393
3.41 FVP_Base_Cortex-A72x2 3-403
3.42 FVP_Base_Cortex-A72x2-A53x4 3-411
3.43 FVP_Base_Cortex-A72x4 3-421
3.44 FVP_Base_Cortex-A72x4-A53x4 3-430
3.45 FVP_Base_Cortex-A73x1 3-441
3.46 FVP_Base_Cortex-A73x1-A53x1 3-449
3.47 FVP_Base_Cortex-A73x2 3-459
3.48 FVP_Base_Cortex-A73x2-A53x4 3-467
3.49 FVP_Base_Cortex-A73x4 3-477
3.50 FVP_Base_Cortex-A73x4-A53x4 3-486
3.51 FVP_Base_Cortex-A75 3-497
3.52 FVP_Base_Cortex-A75x1 3-506
3.53 FVP_Base_Cortex-A75x2 3-514

3.54 FVP_Base_Cortex-A75x4 3-522
3.55 FVP_Base_Cortex-A76AEx2 3-531
3.56 FVP_Base_Cortex-A76AEx4 3-540
3.57 FVP_Base_Cortex-A76x1 3-550
3.58 FVP_Base_Cortex-A76x2 3-558
3.59 FVP_Base_Cortex-A76x4 3-566
3.60 FVP_Base_Cortex-A77x1 3-575
3.61 FVP_Base_Cortex-A77x2 3-583
3.62 FVP_Base_Cortex-A77x4 3-591
3.63 FVP_Base_Cortex-A78Cx1 3-600
3.64 FVP_Base_Cortex-A78Cx2 3-608
3.65 FVP_Base_Cortex-A78Cx4 3-617
3.66 FVP_Base_Cortex-A78x1 3-626
3.67 FVP_Base_Cortex-A78x2 3-634
3.68 FVP_Base_Cortex-A78x4 3-642
3.69 FVP_Base_Cortex-X1x1 3-651
3.70 FVP_Base_Cortex-X1x2 3-659
3.71 FVP_Base_Cortex-X1x4 3-667
3.72 FVP_Base_Neoverse-E1x1 3-676
3.73 FVP_Base_Neoverse-E1x2 3-684
3.74 FVP_Base_Neoverse-E1x4 3-693
3.75 FVP_Base_Neoverse-N1x1 3-702
3.76 FVP_Base_Neoverse-N1x2 3-710
3.77 FVP_Base_Neoverse-N1x4 3-719

Chapter 4 BaseR Platform FVPs

4.1 FVP_BaseR_Cortex-R52x1 4-729
4.2 FVP_BaseR_Cortex-R52x2 4-737
4.3 FVP_BaseR_Cortex-R52x4 4-746

Chapter 5 VE Platform FVPs

5.1 FVP_VE_Cortex-A15x1 5-757
5.2 FVP_VE_Cortex-A15x1-A7x1 5-765
5.3 FVP_VE_Cortex-A15x2 5-774
5.4 FVP_VE_Cortex-A15x2-A7x2 5-782
5.5 FVP_VE_Cortex-A15x4 5-792
5.6 FVP_VE_Cortex-A15x4-A7x4 5-801
5.7 FVP_VE_Cortex-A17x1 5-812
5.8 FVP_VE_Cortex-A17x1-A7x1 5-820
5.9 FVP_VE_Cortex-A17x2 5-829
5.10 FVP_VE_Cortex-A17x4 5-837
5.11 FVP_VE_Cortex-A17x4-A7x4 5-846
5.12 FVP_VE_Cortex-A5x1 5-857
5.13 FVP_VE_Cortex-A5x2 5-865
5.14 FVP_VE_Cortex-A5x4 5-873
5.15 FVP_VE_Cortex-A7x1 5-882
5.16 FVP_VE_Cortex-A7x2 5-890
5.17 FVP_VE_Cortex-A7x4 5-898
5.18 FVP_VE_Cortex-A9x1 5-907
5.19 FVP_VE_Cortex-A9x2 5-915

5.20 FVP_VE_Cortex-A9x4 5-923
5.21 FVP_VE_Cortex-R4 5-932
5.22 FVP_VE_Cortex-R5x1 5-940
5.23 FVP_VE_Cortex-R5x2 5-948
5.24 FVP_VE_Cortex-R7x1 5-956
5.25 FVP_VE_Cortex-R7x2 5-963
5.26 FVP_VE_Cortex-R8x1 5-970
5.27 FVP_VE_Cortex-R8x2 5-977
5.28 FVP_VE_Cortex-R8x4 5-984

Chapter 6 MPS2 Platform FVPs

6.1 FVP_MPS2_Cortex-M0 6-993
6.2 FVP_MPS2_Cortex-M0plus 6-1007
6.3 FVP_MPS2_Cortex-M23 6-1022
6.4 FVP_MPS2_Cortex-M3 6-1037
6.5 FVP_MPS2_Cortex-M33 6-1051
6.6 FVP_MPS2_Cortex-M35P 6-1066
6.7 FVP_MPS2_Cortex-M4 6-1081
6.8 FVP_MPS2_Cortex-M55 6-1095
6.9 FVP_MPS2_Cortex-M7 6-1110

Chapter 7 Arm® Neoverse™ reference design FVPs

7.1 About the FVPs 7-1125
7.2 Block diagrams for RD-N1-E1-edge 7-1126
7.3 FVP peripherals 7-1128

Preface

This preface introduces the *Fast Models Fixed Virtual Platforms (FVP) Reference Guide*.

It contains the following:

- *About this book* on page 9.

About this book

Arm® Fixed Virtual Platform Reference. This manual introduces the Fixed Virtual Platforms, and describes how you can use them with other tools.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction
This chapter introduces the document.

Chapter 2 Getting Started with Fixed Virtual Platforms
This chapter describes how to use FVPs.

Chapter 3 Base Platform FVPs
This chapter lists the Base Platform FVPs and the instances in them.

Chapter 4 BaseR Platform FVPs
This chapter lists the BaseR Platform FVPs and the instances in them.

Chapter 5 VE Platform FVPs
This chapter lists the VE Platform FVPs and the instances in them.

Chapter 6 MPS2 Platform FVPs
This chapter lists the MPS2 Platform FVPs and the instances in them.

Chapter 7 Arm® Neoverse™ reference design FVPs
This chapter describes the Arm Neoverse™ N1 edge and Arm Neoverse E1 edge reference design FVPs. These FVPs are collectively referred to as RD-N1-E1-edge FVPs.

Glossary

The Arm® Glossary is a list of terms used in Arm documentation, together with definitions for those terms. The Arm Glossary does not contain terms that are industry standard unless the Arm meaning differs from the generally accepted meaning.

See the *Arm® Glossary* for more information.

Typographic conventions

italic
Introduces special terminology, denotes cross-references, and citations.

bold
Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.

`monospace`
Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.

`monospace`
Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.

`monospace italic`
Denotes arguments to monospace text where the argument is to be replaced by a specific value.

`monospace bold`
Denotes language keywords when used outside example code.

`<and>`
: Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example:

```
MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>
```

SMALL CAPITALS
: Used in body text for a few terms that have specific technical meanings, that are defined in the *Arm® Glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Feedback

Feedback on this product

If you have any comments or suggestions about this product, contact your supplier and give:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

Feedback on content

If you have comments on content then send an e-mail to *errata@arm.com*. Give:

- The title *Fast Models Fixed Virtual Platforms (FVP) Reference Guide*.
- The number 100966_1113_00_en.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

Arm also welcomes general suggestions for additions and improvements.

Note

Arm tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

Other information

- *Arm® Developer*.
- *Arm® Documentation*.
- *Technical Support*.
- *Arm® Glossary*.

Chapter 1
Introduction

This chapter introduces the document.

It contains the following sections:

- *1.1 About FVPs* on page 1-12.
- *1.2 Data collection in FVPs* on page 1-13.

1.1 About FVPs

Fixed Virtual Platforms (FVPs) enable software development without the need for real hardware.

FVPs are available for Linux and Windows hosts, either:

- As pre-built executables. Their composition is fixed, although you can configure their behavior using parameters.
- As source code in the Fast Models product, with the tools required to customize and build them.

Arm provides different types of FVP, based on the following platforms:

- Armv8-A Base Platform.
- Armv8-R BaseR Platform.
- Arm Versatile™ Express development boards.
- Arm MPS2 or Arm MPS2+ platforms, for Cortex®-M series processors.

FVPs are available for all Cortex-A, Cortex-R, and Cortex-M processors, and they support the CADI, MTI, and Iris interfaces, so can be used for debugging and for trace output.

There are several freely available, pre-built Armv8-A FVPs for download from *Arm Ecosystem Models* on Arm Developer which can be used without a license:

- Foundation Platform. This is a basic FVP with a minimal peripheral set which is suitable for running on Linux hosts only. It includes a single cluster which can be configured with between 1-4 AEMv8-A cores. It is suitable for running bare-metal applications and for booting Linux. It supports the CADI debug interface, but does not support MTI or Iris interfaces.
- AEMv8-A Base Platform RevC FVP. This is a platform model with a more extended peripheral set than the Foundation Platform. It has two AEMv8-A clusters, each of which can be configured with 1-4 AEMv8-A cores. It supports Armv8-A architecture versions up to v8.6 and supports CADI, MTI, and Iris debug and trace interfaces.
- Armv8-A Compliance FVP. This FVP is optimized for validating CPU implementations and can be used with the A-profile Architecture Compliance Kit (ACK) to demonstrate compliance with the Arm architecture specification.

Arm provides validated Linux and Android deliverables for the Armv8-A AEM Base Platform FVP and for the Foundation Platform. These are available on the *Arm Development Platforms wiki* on Arm Community. To get started with Linux on Armv8-A FVPs, see *FVPs* on Arm Community.

Related information

Base Platform

Microcontroller Prototyping System 2

Versatile Express Model

1.2 Data collection in FVPs

Arm periodically collects anonymous information about the usage of our products to understand and analyze what components or features you are using, with the goal of improving our products and your experience with them. Product usage analytics contain information such as system information, settings, and usage of specific features of the product. They do not include any personal information.

Host information includes:

- Operating system name, version, and locale.
- Number of CPUs.
- Amount of physical memory.
- Screen resolution.
- Processor and GPU type.

Feature tracking information includes:

Table 1-1 Fast Models analytics data points

Name	Description	Since
Platform name	• Tracked: — Name of the platform model being run — Fast Models version and build number that was used to build the platform model. — Whether the platform model is a standalone product or was supplied as part of the Fast Models product. [a] • Reported: Percentage of users using the different platforms. • Data type: Text. • Send policy: Every invocation. • Trigger points: On starting the simulation.	v11.8
Session length	• Tracked: Length of time the platform was used. • Reported: Average time the different platforms are used. • Data type: Text. • Send policy: Every invocation. • Trigger points: On exiting the simulation.	v11.9
Debug server	• Tracked: Whether a CADI or Iris debugger was connected. • Reported: Type of debug server that was started, either CADI or Iris. • Data type: Text. • Send policy: Every invocation. • Trigger points: Debug server startup.	v11.10
Arm IP	• Tracked: Names of Fast Models core, System IP, and GPU components that are included in the simulation. • Reported: Component names, for example DP500 or SMMUv3AEM. These are not hierarchical names. • Data type: Text. • Send policy: Every invocation. • Trigger points: Component instantiation.	v11.10

[a] This feature is tracked since v11.10.

Note

- Analytics gathering is enabled by default. Use the `--disable-analytics` command-line option to disable it for the current invocation, or set the `ARM_DISABLE_ANALYTICS` environment variable to a non-zero value to disable it for all invocations.
- Querying the model, for example with `--list-params` or `--help`, does not trigger reporting.
- The names of non-Arm platforms or components are obfuscated.

Chapter 2
Getting Started with Fixed Virtual Platforms

This chapter describes how to use FVPs.

It contains the following sections:

- *2.1 Contents of the package* on page 2-16.
- *2.2 FVP command-line options* on page 2-17.
- *2.3 Loading and running an application on an FVP* on page 2-22.
- *2.4 Configuring the model* on page 2-23.
- *2.5 FVP debug* on page 2-24.
- *2.6 Using the CLCD window* on page 2-25.
- *2.7 Ethernet with VE FVPs* on page 2-28.
- *2.8 Using a terminal with a system model* on page 2-30.
- *2.9 Virtio P9 device component* on page 2-33.

2.1 Contents of the package

The FVP Standard Library consolidates commonly used FVPs into a single package which also contains some useful plug-ins and utilities.

Note

The package does not include unlicensed FVPs. These are available for download separately, from *Arm Ecosystem Models* on Arm Developer.

The package installs the FVPs under the `models` directory. It also installs:

Plug-ins
Plug-ins are DLLs or shared objects that provide extra functionality to FVPs, for instance different types of trace output. To load a plug-in, pass the name of the plug-in to the FVP at startup using the `--plugin` command-line option or using the `FM_TRACE_PLUGINS` environment variable. For more information about plug-ins, see *Plug-ins for Fast Models* in the Fast Models Reference Manual.

Model Shell and Model Debugger
Model Shell is a command-line tool for launching FVPs. For more information about Model Shell, see *Model Shell for Fast Models Reference Manual*. Model Debugger is an easy to use symbolic debugger with a GUI that allows you to debug software running on the FVP. For more information about Model Debugger, see *Model Debugger for Fast Models User Guide*. They are installed in the `bin` directory.

iris.debug Python module
iris.debug is a Python scripting interface to Fast Models. It allows you to interact with FVPs, including connecting to and configuring them, performing execution control, and accessing registers and memory. It is installed under the `Iris` directory. For more information about iris.debug, see *Iris Python Debug Scripting User Guide*.

2.2 FVP command-line options

Specify these options when you launch an FVP from the command line. You can specify these options in any order.

Table 2-1 CADI or Iris-related

Short form	Long form	Description
`-S`	`--cadi-server`	Start a CADI server. This option allows a CADI-enabled debugger to connect to targets in the simulation. To shut down the server, return to the command window that you used to start the model and press **Ctrl+C**.
`-I`	`--iris-server`	Start an Iris server. This option allows an Iris-enabled debugger to connect to targets in the simulation.
`-R`	`--run`	Run the simulation immediately after the CADI or Iris server is started. Use this option with `--cadi-server` or `--iris-server`. The default is to wait until the debugger has connected before running.
`-L`	`--cadi-log`	Log all CADI function calls made during the simulation into XML files. One log file is created for each CADI target. The log files are created in the current working directory. The filename format is: `CADIlog-<TargetInstanceName>-<ProcessID>.xml`
`-i`	`--iris-log`	Log to stdout all Iris function calls that were made during the simulation. There are 5 possible log levels. To set a level greater than 1, specify the option multiple times, for example `-ii` for level 2. The log levels have the following meanings: **0** Logging is disabled. This value is the default. **1** Log messages use a compact single-line format. **2** Log messages use a single-line pseudo-JSON format. **3** Log messages use a more readable, multi-line pseudo-JSON format. **4** As 3 but also prints the U64JSON hex value of the message. **Note** To set the Iris log level for all FVP invocations, use the `IRIS_GLOBAL_INSTANCE_LOG_MESSAGES` environment variable.
`-A`	`--iris-allow-remote`	Start an Iris server and allow connections to it from a remote workstation. The default is disallowed.

Table 2-1 CADI or Iris-related (continued)

Short form	Long form	Description
`-p`	`--print-port-number`	Print the port number on which the Iris or CADI server is listening. Use this option with `--cadi-server` or `--iris-server`. **Tip** This option can be useful if you need to specify the port number when you connect a client to the debug server.
	`--iris-port` *n*	Set a port to use for the Iris server. Use this option with `--iris-server`. The default is 7100.
	`--iris-port-range` *min*`:`*max*	Set the range of ports to scan when starting an Iris server. The server uses the first available port in the range. Use this option with `--iris-server`.

Table 2-2 Output

Short form	Long form	Description
	`--list-instances`	Print a list of model instances to standard output, then exit the simulation. Use this option to help identify the correct syntax for configuration files, and to find out what instances the target supplies.
`-l`	`--list-params`	Print a list of model parameters to standard output, then exit the simulation. **Tip** If you are loading a plug-in, this option also lists the plug-in parameters.
	`--dump-params`	Dump the list of model parameters into a JSON file called `parameter_list.json`, then exit the simulation. The file is created in the current working directory.
	`--list-regs`	Print model register information to standard output, then exit the simulation.
	`--check-regs`	Same as `--list-regs` but with extra consistency checks on the CADI register API.
`-o`	`--output` *filename*	Redirect output from the `--list-instances`, `--list-memory`, `--list-params`, and `--list-regs` commands to a file. If this option is used with `--list-params`, the contents of the output file are formatted correctly for use as input by the `--config-file` option.
	`--log` *filename*	Log all SystemC reports into *filename*.

Table 2-2 Output (continued)

Short form	Long form	Description
	`--stat`	Print the following performance statistics on simulation exit: **Simulated time** An estimate of the time that the workload would have taken on the modeled hardware. **User time** Time in wall clock seconds that the host CPU spent running in user mode. **System time** Time in wall clock seconds that the host CPU spent running in system mode. **Wall time** Time in wall clock seconds between the simulation starting and stopping. **Performance index** An estimate of the accuracy of the simulation performance. This value is Simulated time divided by Wall time.
`-P`	`--prefix`	Prefix each line of semihosting output with the name of the target instance.
`-h`	`--help`	Print the help message and exit.
	`--version`	Print version information for the FVP.
`-q`	`--quiet`	Suppress informational output.
`-K`	`--keep-console`	Keep the console window open after completion. This option applies to Windows only.
	`--disable-model-exitcode`	Disable the simulation from retrieving the exit code returned by a model or a plug-in. By default, it is enabled.

Table 2-3 Run control

Short form	Long form	Description
	`--cpulimit` *n*	Maximum number of wall-clock seconds for the simulation process to be active. This value excludes simulation startup and shutdown. This option is ignored if a debug server is started. The default is unlimited.
	`--cyclelimit` *n*	Maximum number of cycles to run. This option is ignored if a debug server is started. The default is unlimited.
`-T`	`--timelimit` *n*	Maximum number of wall-clock seconds for the simulation to run, excluding startup and shutdown. To terminate the model immediately after initialization, specify `--timelimit 0`.

Table 2-3 Run control (continued)

Short form	Long form	Description
	`--simlimit` *n*	Maximum number of seconds to simulate. This option is ignored if a debug server is started. The default is unlimited. Like the `Simulated time` value output by `--stat`, this value is measured in simulation seconds, not wall-clock seconds.
`-b`	`--break` [*instance*=] *address*	Set a program breakpoint on the address of an instruction. This option can be specified multiple times. If the FVP has multiple cores you must specify an instance, for example: `-b FVP_Base_AEMv8A.cluster0.cpu0=0x800010eC`

Table 2-4 Timing and performance

Short form	Long form	Description
	`--cpi-file` *filename*	Use *filename* to set the Cycles Per Instruction (CPI) class. For information about CPI files, see *Timing Annotation*.
`-Q`	`--quantum` *n*	Number of ticks to simulate for each quantum. The default is 10000.
`-M`	`--min-sync-latency` *n*	Number of ticks to simulate before synchronizing. Events that occur at a higher frequency than this value are missed. The default is 100.
	`--fast-ram` *filename*	Enable FastRAM and load the configuration from *filename*. For more information about FastRAM, see *FastRAM*.

Table 2-5 Memory

Short form	Long form	Description
	`--dump` *file@address,size*	Dump a section of memory to a file at model shutdown. This option can be specified multiple times. The full syntax is: `--dump [instance=]file@[memspace:]address,size` **Tip** To see the list of instances and memory spaces, use the `--list-memory` option.
	`--data` *file@address*	Write raw data contained in *file* to the specified address. This option can be specified multiple times. The full syntax is: `--data [instance=]file@[memspace:]address`

Table 2-5 Memory (continued)

Short form	Long form	Description
	`--list-memory`	Print model memory information to standard output, then exit the simulation.
	`--start` [*instance*=] *address*	Set the initial PC value to this address, overriding the `.axf` start address. **Note** • Use this option if you do not want the CPU to start executing at the default reset address. You do not normally need to do this if you are loading an ELF file using `--application`. • This option can be used with `--data` to load binary data that is not in an ELF file.

Table 2-6 Configuration

Short form	Long form	Description
`-C`	`--parameter` *instance.parameter=value*	Set a parameter. This option can be specified multiple times. Specify the full hierarchical name of the parameter. This option is also used to set plug-in parameters.
`-f`	`--config-file` *filename*	Load the parameters from a configuration file.

Table 2-7 Options for loading a plug-in or application

Short form	Long form	Description
`-a`	`--application` [*instance*=] *filename*	Load an application. On a multi-core system, specify the instance, or use * to load the application image into all the cores in a cluster: `-a cluster0.cpu*=file`
	`--plugin` *filename*	Load the plug-in *filename*. This option can be specified multiple times. You can also load plug-ins using the `FM_TRACE_PLUGINS` environment variable. For information about plug-ins, see *Plug-ins for Fast Models*.
	`--trace-plugin` *filename*	Load a trace plug-in. **Note** This option is deprecated. Use `--plugin` instead.

Table 2-8 Analytics

Short form	Long form	Description
	`--disable-analytics`	Disable product analytics gathering for the current run. For persistent disabling, set the `ARM_DISABLE_ANALYTICS` environment variable.

2.3 Loading and running an application on an FVP

There are different ways to launch an FVP, for example from the command prompt, or from Model Debugger or Arm Development Studio.

To run an FVP from the command prompt, enter the model name followed by the model options. To see all available options, use the `--help` option. This is a list of some of the commonly used options for FVPs:

`-a [`*`instance=`*`]`*`filename`*`.axf`
: Specifies an application to load, and optionally, the instance or instances to load it on. The file can be in one of the following formats, or in a gzip-compressed version:
- ELF.
- Motorola S-Record.
- Intel-Hex.
- Verilog-Hex, in the format:

```
@<address_in_hex> <byte_in_hex>
```

If the FVP contains multiple core instances, you can specify the instance to load the image on. The instance name can include a wildcard (*) to load the same application image into multiple cores, for example:

```
FVP_Base_AEMv8A-AEMv8A -a cluster0.cpu*=__image.axf
```

Omitting the instance name loads the application on all cores in the first cluster. If the FVP has multiple cores but no clusters, you must specify the instance name.

`--data `*`filename`*`.bin@`*`address`*
: Loads binary data into memory at the address specified.

`-C `*`instance.parameter=value`*
: Sets a single model parameter. Parameters are specified using a path that separates the instance names and the parameter using dots. For example, `-C bp.flashloader0.fname=fip.bin`. Here, `bp` and `flashloader0` are instance names and `fname` is the parameter. To set multiple parameters using a configuration file, use the `-f` option instead. To list all the available parameters, with their type, default value, and description, run the model with the `--list-params`, or `-l` option.

`-f `*`config_file`*`.txt`
: Specifies the name of a plain text configuration file. Configuration files simplify managing multiple model parameters. You can set the same parameters using this option as with the `-C` option.

`-S`
: Starts a CADI debug server. This option allows a CADI-enabled debugger, such as Model Debugger or Arm Development Studio Debugger, to connect to the running model. By default, the model waits for the debugger to connect before starting.

Related information

Arm Development Studio User Guide

Model Debugger for Fast Models User Guide

2.4 Configuring the model

When you start the model from the command line, you can configure it using either:

- One or more `-C` command-line arguments.
- A configuration file and the `-f` command-line argument.

Each `-C` command-line argument or line in the configuration file must contain:

- The name of the component instance.
- The parameter to modify.
- Its value.

Use the following format:

`instance.parameter=value`

The `instance` can be a hierarchical path, with each level separated by a dot `.` character.

Note

- Comment lines in the configuration file begin with a `#` character.
- You can set Boolean values using either `true` or `false`, or `1` or `0`.

You can generate a configuration file with all parameters set to default values by using the `-o` option to redirect the output from the `--list-params` option, for example:

```
FVP_Base_AEMv8A.exe --list-params -o params.txt
```

2.5 FVP debug

This section describes how to debug an FVP.

FVP debug options

To debug an FVP, you can either:

- Run the FVP from within a CADI-enabled debugger.
- Start the FVP with the `-S` command-line argument and then connect a CADI-enabled debugger to it.

For information about using your debugger in these ways, see your debugger documentation.

Semihosting support

Semihosting enables code running on a platform model to directly access the I/O facilities on a host computer. Examples of these facilities include console I/O and file I/O.

The simulator handles semihosting by intercepting `HLT 0xF000`, `SVC 0x123456`, or `SVC 0xAB`, depending on whether the processor is in A64, A32 or T32 state. It handles all other `HLT`s and `SVC`s as normal.

If the operating system does not use `HLT 0xF000`, `SVC 0x123456`, or `SVC 0xAB` for its own purposes, it is not necessary to disable semihosting support to boot an operating system.

To temporarily or permanently disable semihosting support for a current debug connection, see your debugger documentation.

Related information

Semihosting for AArch32 and AArch64

Using semihosting to access resources on the host computer

2.6 Using the CLCD window

When a Base Platform or VE FVP starts, the CLCD window opens, representing the contents of the simulated color LCD frame buffer. It automatically resizes to match the horizontal and vertical resolution that is set in the CLCD peripheral registers.

Figure 2-1 CLCD window in its default state at startup

The top section of the CLCD window displays the status information.

USERSW

Eight white boxes show the state of the User DIP switches.

These represent switch S6 on the VE hardware, USERSW[8:1], which is mapped to bits [7:0] of the SYS_SW register at address `0x1C010004`.

The switches are in the off position by default. To change its state, click in the area above or below a white box.

BOOTSW

Eight white boxes show the state of the VE Boot DIP switches.

These represent switch S8 on the VE hardware, BOOTSEL[8:1], which is mapped to bits [15:8] of the SYS_SW register at address `0x1C010004`.

The switches are in the off position by default.

——— **Note** ———

Changing Boot DIP switch positions while the model is running can result in unpredictable behavior.

———————————

S6LED

Eight colored boxes indicate the state of the VE User LEDs.

These represent the red/yellow/green LEDs on the VE hardware, which are mapped to bits [7:0] of the SYS_LED register at address `0x1C010008`.

Daughter

Eight white boxes show the state of the daughterboard DIP switches and eight colored boxes show the state of the daughterboard LEDs.

Total Instr

A counter showing the total number of instructions executed.

Because the FVP models provide a *Programmer's View* (PV) of the system, the CLCD displays total instructions rather than total processor cycles. Timing might differ substantially from the hardware because:

- Bus fabric is simplified.
- Memory latencies are minimized.
- Cycle approximate processor and peripheral models are used.

In general, bus transaction timing is consistent with the hardware, but the timing of operations within the model is not accurate.

Total Time

A counter showing the total elapsed time, in seconds.

This time is wall clock time, not simulated time.

Rate Limit

A feature that disables or enables fast simulation.

Because the system model is highly optimized, your code might run faster than it would on real hardware. This effect might cause timing issues.

Rate Limit is enabled by default. Simulation time is restricted so that it more closely matches real time.

To disable or enable Rate Limit, click the square button. You can configure this option when instantiating the model with the `rate_limit-enable` visualization component parameter.

When you click the **Total Instr** item in the CLCD, the display toggles to show the following:

Instr/sec

The number of instructions that execute per second of wall clock time.

Perf Index

The ratio of real time to simulation time. The larger the ratio, the faster the simulation runs. If you enable the Rate Limit feature, the Perf Index approaches unity.

You can reset the simulation counters by resetting the model.

The FVP CLCD displays the core run state for each core on each cluster using a colored icon. The icons are to the left of the **Total Instr** (or **Inst/sec**) item.

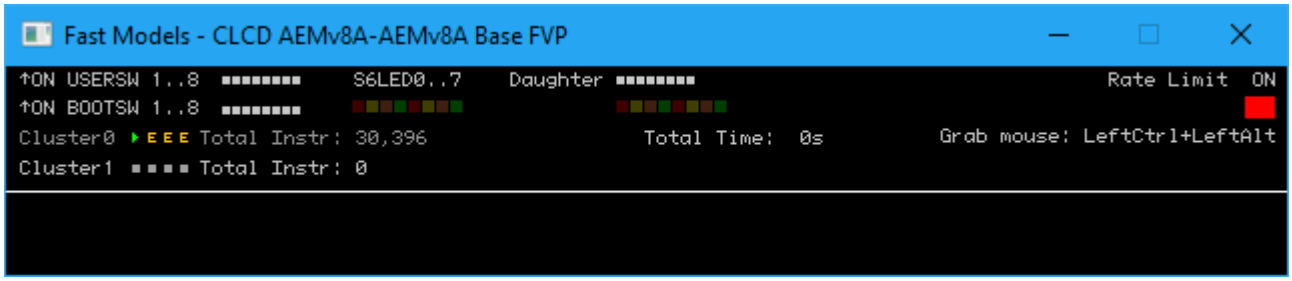

Figure 2-2 Core run state icons for a dual-cluster, quad-core model

Table 2-9 Core run state icon descriptions

Icon	State label	Description
	UNKNOWN	Run status unknown, that is, simulation has not started.
	RUNNING	Core running, is not idle, and is executing instructions.
	HALTED	External halt signal asserted.
	STANDBY_WFE	Last instruction executed was WFE and standby mode has been entered.
	STANDBY_WFI	Last instruction executed was WFI and standby mode has been entered.
	IN_RESET	External reset signal asserted.
	DORMANT	Partial core power down.
	SHUTDOWN	Complete core power down.

If the CLCD window has focus:

- Any keyboard input is translated to PS/2 keyboard data.
- Any mouse activity over the window is translated into PS/2 relative mouse motion data. The data is then streamed to the KMI peripheral model FIFOs.

Note

The simulator only sends relative mouse motion events to the model. As a result, the host mouse pointer does not necessarily align with the target OS mouse pointer.

You can hide the host mouse pointer by pressing the **left Ctrl**+**left Alt** keys. Press the keys again to redisplay the host mouse pointer. Only the **left Ctrl** key is operational. The **right Ctrl** key does not have the same effect.

If you prefer to use a different key, configure it with the `trap_key` visualization component parameter.

Related information

VEVisualisation component

2.7 Ethernet with VE FVPs

This section describes how to use Ethernet with VE FVPs.

Using Ethernet with VE FVPs

The VE FVPs have a virtual Ethernet component. This component is a model of the SMSC 91C111 Ethernet controller, and uses a TAP device to communicate with the network. By default, the Ethernet component is disabled.

Host requirements

Before you can use the Ethernet capability of VE FVPs, set up your host computer.

Target requirements

This section describes the target requirements.

Target requirements - about

The VE FVPs include a software implementation of the SMSC 91C111 Ethernet controller. Your target OS must therefore include a driver for this specific device. To use the SMSC chip, configure the kernel. Linux supports the SMSC 91C111.

The configurable SMSC 91C111 component parameters are:

- `enabled`.
- `mac_address`.
- `promiscuous`.

enabled

When the device is disabled, the kernel cannot detect the device.

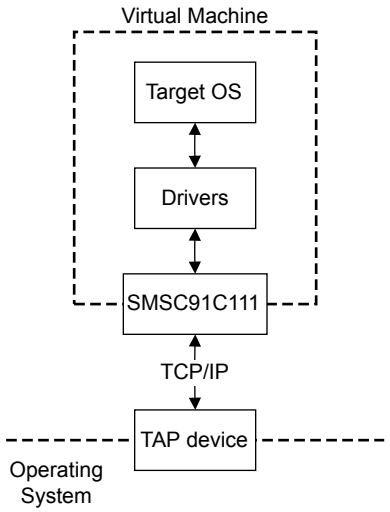

Figure 2-3 Model networking structure block diagram

To perform read and write operations on the TAP device, configure a HostBridge component. The HostBridge component is a virtual *Programmer's View* (PV) model. It acts as a networking gateway to exchange Ethernet packets with the TAP device on the host, and to forward packets to NIC models.

mac_address

There are two options for the `mac_address` parameter.

If a MAC address is not specified, when the simulator is run it takes the default MAC address, which is randomly generated. This random generation provides some degree of MAC address uniqueness when running models on multiple hosts on a local network.

promiscuous

The Ethernet component starts in promiscuous mode by default. In this mode, it receives all network traffic, even any not addressed to the device. Use this mode if you are using a single network device for multiple MAC addresses. Use this mode if, for example, you share the network card between your host OS and the VE FVP Ethernet component.

By default, the Ethernet device on the VE FVP has a randomly generated MAC address and starts in promiscuous mode.

2.8 Using a terminal with a system model

The Terminal component is a virtual component that enables UART data to be transferred between a TCP/IP socket on the host and a serial port on the target.

Installing Telnet on Microsoft Windows 10

Microsoft Windows 10 disables the Telnet client by default. Follow these steps to enable it:

1. Select **Start** > **Settings**.
2. In the search box, enter **Turn Windows features on or off**. The **Windows Features** dialog opens.
3. Select the **Telnet Client** check box and click **OK**. The installation might take several minutes to complete.

The following figure shows a block diagram of one possible relationship between the target and host through the Terminal component. The TelnetTerminal block is what you configure when you define Terminal component parameters. The Virtual Machine is your FVP.

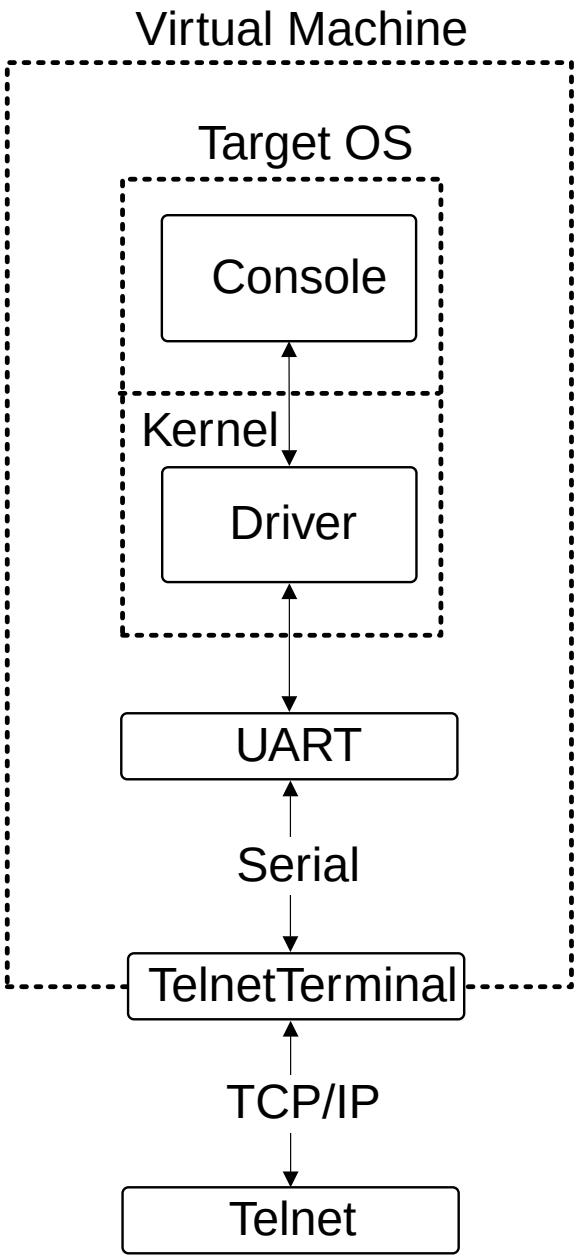

Figure 2-4 Terminal block diagram

On the target side, the console process that is invoked by your target OS relies on a suitable driver being present. Such drivers are normally part of the OS kernel. The driver passes serial data through a UART. The data is forwarded to the TelnetTerminal component, which exposes a TCP/IP port to the world outside of the FVP. This port can be connected to by, for example, a Telnet process on the host.

By default, the FVP starts four telnet Terminals when the model is initialized. You can change the startup behavior for each of the four Terminals by modifying the corresponding component parameters.

If the Terminal connection is broken, for example by closing a client telnet session, the port is re-opened on the host. This might have a different port number if the original one is no longer available. Before the first data access, you can connect a client of your choice to the network socket. If there is no existing connection when the first data access is made, and the `start_telnet` parameter is `true`, a host telnet session is started automatically.

The port number of a particular Terminal instance can be defined when the FVP starts. The actual value of the port that is used by each Terminal is declared when it starts or restarts, and might not be the value that you specified if the port is already in use. If you are using Model Shell, the port numbers are displayed in the host window in which you started the model.

You can start the Terminal component in either telnet mode or raw mode.

Telnet mode

In telnet mode, the Terminal component supports a subset of the RFC 854 protocol. This means that the Terminal participates in negotiations between the host and client concerning what is and is not supported, but flow control is not implemented.

Raw mode

Raw mode enables the byte stream to pass unmodified between the host and the target. This means that the Terminal component does not participate in initial capability negotiations between the host and client. It acts as a TCP/IP port. You can use this feature to directly connect to your target through the Terminal component.

2.9 Virtio P9 device component

The VirtioP9Device component is included in Base, BaseR, and A-profile VE platforms. It implements a subset of the Plan 9 file protocol over a virtio transport. It enables accessing a directory on the host's filesystem within Linux, or another operating system that implements the protocol, running on a platform model.

Take the following steps to set up this component:

- Use a version of Linux that supports v9fs over virtio and virtio-mmio devices.
- Update the device tree to include the VirtioP9Device component, or specify it on the kernel command-line, as shown below. The address range for both VE and Base platforms is `0x1C140000-0x1C14FFFF`.

  The interrupt number is 43, or IRQ 75, for both VE and Base platforms.
- Set the following parameter to the directory on the host that you want to mount in the model:

  VE:

  `motherboard.virtiop9device.root_path`

  Base:

  `bp.virtiop9device.root_path`
- On Linux, mount the host directory by using the following command in the model:

```
$ mount -t 9p -o trans=virtio,version=9p2000.L FM <mount point>
```

Example kernel command-line argument:

```
virtio_mmio.device=0x10000@0x1c140000:75
```

Example entry for DTS files, to add next to the corresponding `virtio_block` entry:

```
virtio_p9@0140000 {
       compatible = "virtio,mmio";
       reg = <0x0 0x1c140000 0x0 0x1000>;
       interrupts = <0x0 0x2b 0x4>;
};
```

Chapter 3
Base Platform FVPs

This chapter lists the Base Platform FVPs and the instances in them.

For the Base Platform memory map, see *Base Platform memory map* in the Fast Models Reference Manual.

It contains the following sections:

- *3.1 FVP_Base_AEMv8A-AEMv8A* on page 3-36.
- *3.2 FVP_Base_Cortex-A32x1* on page 3-47.
- *3.3 FVP_Base_Cortex-A32x2* on page 3-55.
- *3.4 FVP_Base_Cortex-A32x4* on page 3-63.
- *3.5 FVP_Base_Cortex-A35x1* on page 3-72.
- *3.6 FVP_Base_Cortex-A35x2* on page 3-80.
- *3.7 FVP_Base_Cortex-A35x4* on page 3-88.
- *3.8 FVP_Base_Cortex-A53x1* on page 3-97.
- *3.9 FVP_Base_Cortex-A53x2* on page 3-105.
- *3.10 FVP_Base_Cortex-A53x4* on page 3-113.
- *3.11 FVP_Base_Cortex-A55* on page 3-122.
- *3.12 FVP_Base_Cortex-A55+Cortex-A76* on page 3-131.
- *3.13 FVP_Base_Cortex-A55x1* on page 3-140.
- *3.14 FVP_Base_Cortex-A55x1+Cortex-A75x1* on page 3-148.
- *3.15 FVP_Base_Cortex-A55x2* on page 3-157.
- *3.16 FVP_Base_Cortex-A55x2+Cortex-A75x2* on page 3-165.
- *3.17 FVP_Base_Cortex-A55x4* on page 3-174.
- *3.18 FVP_Base_Cortex-A55x4+Cortex-A75x1* on page 3-183.
- *3.19 FVP_Base_Cortex-A55x4+Cortex-A75x2* on page 3-192.
- *3.20 FVP_Base_Cortex-A55x4+Cortex-A75x4* on page 3-202.
- *3.21 FVP_Base_Cortex-A55x4+Cortex-A76x2* on page 3-212.

- *3.22 FVP_Base_Cortex-A57x1* on page 3-222.
- *3.23 FVP_Base_Cortex-A57x1-A35x1* on page 3-230.
- *3.24 FVP_Base_Cortex-A57x1-A53x1* on page 3-240.
- *3.25 FVP_Base_Cortex-A57x2* on page 3-250.
- *3.26 FVP_Base_Cortex-A57x2-A35x4* on page 3-258.
- *3.27 FVP_Base_Cortex-A57x2-A53x4* on page 3-268.
- *3.28 FVP_Base_Cortex-A57x4* on page 3-278.
- *3.29 FVP_Base_Cortex-A57x4-A35x4* on page 3-287.
- *3.30 FVP_Base_Cortex-A57x4-A53x4* on page 3-298.
- *3.31 FVP_Base_Cortex-A65AEx2* on page 3-309.
- *3.32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2* on page 3-318.
- *3.33 FVP_Base_Cortex-A65AEx4* on page 3-328.
- *3.34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4* on page 3-338.
- *3.35 FVP_Base_Cortex-A65AEx8* on page 3-348.
- *3.36 FVP_Base_Cortex-A65x1* on page 3-359.
- *3.37 FVP_Base_Cortex-A65x2* on page 3-367.
- *3.38 FVP_Base_Cortex-A65x4* on page 3-376.
- *3.39 FVP_Base_Cortex-A72x1* on page 3-385.
- *3.40 FVP_Base_Cortex-A72x1-A53x1* on page 3-393.
- *3.41 FVP_Base_Cortex-A72x2* on page 3-403.
- *3.42 FVP_Base_Cortex-A72x2-A53x4* on page 3-411.
- *3.43 FVP_Base_Cortex-A72x4* on page 3-421.
- *3.44 FVP_Base_Cortex-A72x4-A53x4* on page 3-430.
- *3.45 FVP_Base_Cortex-A73x1* on page 3-441.
- *3.46 FVP_Base_Cortex-A73x1-A53x1* on page 3-449.
- *3.47 FVP_Base_Cortex-A73x2* on page 3-459.
- *3.48 FVP_Base_Cortex-A73x2-A53x4* on page 3-467.
- *3.49 FVP_Base_Cortex-A73x4* on page 3-477.
- *3.50 FVP_Base_Cortex-A73x4-A53x4* on page 3-486.
- *3.51 FVP_Base_Cortex-A75* on page 3-497.
- *3.52 FVP_Base_Cortex-A75x1* on page 3-506.
- *3.53 FVP_Base_Cortex-A75x2* on page 3-514.
- *3.54 FVP_Base_Cortex-A75x4* on page 3-522.
- *3.55 FVP_Base_Cortex-A76AEx2* on page 3-531.
- *3.56 FVP_Base_Cortex-A76AEx4* on page 3-540.
- *3.57 FVP_Base_Cortex-A76x1* on page 3-550.
- *3.58 FVP_Base_Cortex-A76x2* on page 3-558.
- *3.59 FVP_Base_Cortex-A76x4* on page 3-566.
- *3.60 FVP_Base_Cortex-A77x1* on page 3-575.
- *3.61 FVP_Base_Cortex-A77x2* on page 3-583.
- *3.62 FVP_Base_Cortex-A77x4* on page 3-591.
- *3.63 FVP_Base_Cortex-A78Cx1* on page 3-600.
- *3.64 FVP_Base_Cortex-A78Cx2* on page 3-608.
- *3.65 FVP_Base_Cortex-A78Cx4* on page 3-617.
- *3.66 FVP_Base_Cortex-A78x1* on page 3-626.
- *3.67 FVP_Base_Cortex-A78x2* on page 3-634.
- *3.68 FVP_Base_Cortex-A78x4* on page 3-642.
- *3.69 FVP_Base_Cortex-X1x1* on page 3-651.
- *3.70 FVP_Base_Cortex-X1x2* on page 3-659.
- *3.71 FVP_Base_Cortex-X1x4* on page 3-667.
- *3.72 FVP_Base_Neoverse-E1x1* on page 3-676.
- *3.73 FVP_Base_Neoverse-E1x2* on page 3-684.
- *3.74 FVP_Base_Neoverse-E1x4* on page 3-693.
- *3.75 FVP_Base_Neoverse-N1x1* on page 3-702.
- *3.76 FVP_Base_Neoverse-N1x2* on page 3-710.
- *3.77 FVP_Base_Neoverse-N1x4* on page 3-719.

3.1 FVP_Base_AEMv8A-AEMv8A

FVP_Base_AEMv8A-AEMv8A contains the following instances:

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A	FVP_Base_AEMv8A_AEMv8A	Base Platform Compute Subsystem for ARMAEMv8AMPCT and ARMAEMv8AMPCT.
FVP_Base_AEMv8A_AEMv8A.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_AEMv8A_AEMv8A.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_AEMv8A_AEMv8A.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_AEMv8A_AEMv8A.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_AEMv8A_AEMv8A.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_AEMv8A_AEMv8A.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_AEMv8A_AEMv8A.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_AEMv8A_AEMv8A.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_AEMv8A_AEMv8A.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_AEMv8A_AEMv8A.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_AEMv8A_AEMv8A.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_AEMv8A_AEMv8A.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_AEMv8A_AEMv8A.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_AEMv8A_AEMv8A.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_AEMv8A_AEMv8A.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_AEMv8A_AEMv8A.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_AEMv8A_AEMv8A.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_AEMv8A_AEMv8A.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_AEMv8A_AEMv8A.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_AEMv8A_AEMv8A.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_AEMv8A_AEMv8A.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_AEMv8A_AEMv8A.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_AEMv8A_AEMv8A.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_AEMv8A_AEMv8A.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_AEMv8A_AEMv8A.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_AEMv8A_AEMv8A.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_AEMv8A_AEMv8A.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_AEMv8A_AEMv8A.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_AEMv8A_AEMv8A.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_AEMv8A_AEMv8A.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_AEMv8A_AEMv8A.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_AEMv8A_AEMv8A.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_AEMv8A_AEMv8A.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_AEMv8A_AEMv8A.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_AEMv8A_AEMv8A.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_AEMv8A_AEMv8A.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_AEMv8A_AEMv8A.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_AEMv8A_AEMv8A.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_AEMv8A_AEMv8A.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_AEMv8A_AEMv8A.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_AEMv8A_AEMv8A.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_AEMv8A_AEMv8A.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_AEMv8A_AEMv8A.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_AEMv8A_AEMv8A.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_AEMv8A_AEMv8A.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_AEMv8A_AEMv8A.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_AEMv8A_AEMv8A.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_AEMv8A_AEMv8A.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_AEMv8A_AEMv8A.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_AEMv8A_AEMv8A.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_AEMv8A_AEMv8A.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_AEMv8A_AEMv8A.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_AEMv8A_AEMv8A.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_AEMv8A_AEMv8A.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_AEMv8A_AEMv8A.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_AEMv8A_AEMv8A.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_AEMv8A_AEMv8A.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_AEMv8A_AEMv8A.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_AEMv8A_AEMv8A.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_AEMv8A_AEMv8A.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_AEMv8A_AEMv8A.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_AEMv8A_AEMv8A.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_AEMv8A_AEMv8A.cluster0	Cluster_ARMAEMv8-A_MP	ARMAEMv8-A Cluster CT model.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu0	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu1	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu2	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu3	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster0_labeller	*Labeller*	-
FVP_Base_AEMv8A_AEMv8A.cluster1	Cluster_ARMAEMv8-A_MP	ARMAEMv8-A Cluster CT model.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu0	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu1	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu2	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu3	ARMAEMv8-A_MP	ARMAEMv8-A MP CT model.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1.cpu3.l1icache	*PVCache*	PV Cache.

Table 3-1 FVP_Base_AEMv8A-AEMv8A instances (continued)

Name	Type	Description
FVP_Base_AEMv8A_AEMv8A.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_AEMv8A_AEMv8A.cluster1_labeller	*Labeller*	-
FVP_Base_AEMv8A_AEMv8A.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_AEMv8A_AEMv8A.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_AEMv8A_AEMv8A.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_AEMv8A_AEMv8A.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.2 FVP_Base_Cortex-A32x1

FVP_Base_Cortex-A32x1 contains the following instances:

Table 3-2 FVP_Base_Cortex-A32x1 instances

Name	Type	Description
FVP_Base_Cortex_A32x1	FVP_Base_Cortex_A32x1	Base Platform Compute Subsystem for ARMCortexA32x1CT.
FVP_Base_Cortex_A32x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A32x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A32x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A32x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A32x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-2 FVP_Base_Cortex-A32x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-2 FVP_Base_Cortex-A32x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A32x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A32x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A32x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A32x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A32x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A32x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-2 FVP_Base_Cortex-A32x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A32x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A32x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A32x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A32x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A32x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-2 FVP_Base_Cortex-A32x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A32x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A32x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A32x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A32x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A32x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A32x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A32x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-2 FVP_Base_Cortex-A32x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A32x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A32x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A32x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A32x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A32x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A32x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-2 FVP_Base_Cortex-A32x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A32x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A32x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A32x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A32x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A32x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A32x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A32x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A32x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A32x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A32x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A32x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A32x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A32x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A32x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A32x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-2 FVP_Base_Cortex-A32x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A32x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x1.cluster0	Cluster_ARM_Cortex-A32	ARM Cortex-A32 Cluster CT model.
FVP_Base_Cortex_A32x1.cluster0.cpu0	ARM_Cortex-A32	ARM Cortex-A32 CT model.
FVP_Base_Cortex_A32x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A32x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A32x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A32x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A32x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A32x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.3 FVP_Base_Cortex-A32x2

FVP_Base_Cortex-A32x2 contains the following instances:

Table 3-3 FVP_Base_Cortex-A32x2 instances

Name	Type	Description
FVP_Base_Cortex_A32x2	FVP_Base_Cortex_A32x2	Base Platform Compute Subsystem for ARMCortexA32x2CT.
FVP_Base_Cortex_A32x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A32x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A32x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A32x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A32x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-3 FVP_Base_Cortex-A32x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-3 FVP_Base_Cortex-A32x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A32x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A32x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A32x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A32x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A32x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A32x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-3 FVP_Base_Cortex-A32x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A32x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A32x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A32x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A32x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A32x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-3 FVP_Base_Cortex-A32x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A32x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A32x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A32x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A32x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A32x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A32x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A32x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-3 FVP_Base_Cortex-A32x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A32x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A32x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A32x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A32x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A32x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A32x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-3 FVP_Base_Cortex-A32x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A32x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A32x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A32x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A32x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A32x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A32x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A32x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A32x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A32x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A32x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A32x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A32x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A32x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A32x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A32x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-3 FVP_Base_Cortex-A32x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A32x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x2.cluster0	Cluster_ARM_Cortex-A32	ARM Cortex-A32 Cluster CT model.
FVP_Base_Cortex_A32x2.cluster0.cpu0	ARM_Cortex-A32	ARM Cortex-A32 CT model.
FVP_Base_Cortex_A32x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A32x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x2.cluster0.cpu1	ARM_Cortex-A32	ARM Cortex-A32 CT model.
FVP_Base_Cortex_A32x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A32x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x2.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A32x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A32x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A32x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A32x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.4 FVP_Base_Cortex-A32x4

FVP_Base_Cortex-A32x4 contains the following instances:

Table 3-4 FVP_Base_Cortex-A32x4 instances

Name	Type	Description
FVP_Base_Cortex_A32x4	FVP_Base_Cortex_A32x4	Base Platform Compute Subsystem for ARMCortexA32x4CT.
FVP_Base_Cortex_A32x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A32x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A32x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A32x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A32x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A32x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A32x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A32x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A32x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A32x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A32x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A32x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A32x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A32x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A32x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A32x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A32x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A32x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A32x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A32x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A32x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A32x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A32x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A32x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A32x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A32x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A32x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A32x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A32x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A32x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A32x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A32x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A32x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A32x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A32x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A32x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A32x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A32x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A32x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A32x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A32x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A32x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A32x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A32x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A32x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A32x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A32x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A32x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A32x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A32x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A32x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A32x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A32x4.cluster0	Cluster_ARM_Cortex-A32	ARM Cortex-A32 Cluster CT model.
FVP_Base_Cortex_A32x4.cluster0.cpu0	ARM_Cortex-A32	ARM Cortex-A32 CT model.
FVP_Base_Cortex_A32x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A32x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0.cpu1	ARM_Cortex-A32	ARM Cortex-A32 CT model.
FVP_Base_Cortex_A32x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A32x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0.cpu2	ARM_Cortex-A32	ARM Cortex-A32 CT model.
FVP_Base_Cortex_A32x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A32x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0.cpu3	ARM_Cortex-A32	ARM Cortex-A32 CT model.
FVP_Base_Cortex_A32x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A32x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-4 FVP_Base_Cortex-A32x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A32x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A32x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A32x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A32x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A32x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A32x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.5 FVP_Base_Cortex-A35x1

FVP_Base_Cortex-A35x1 contains the following instances:

Table 3-5 FVP_Base_Cortex-A35x1 instances

Name	Type	Description
FVP_Base_Cortex_A35x1	FVP_Base_Cortex_A35x1	Base Platform Compute Subsystem for ARMCortexA35x1CT.
FVP_Base_Cortex_A35x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A35x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A35x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A35x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A35x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-5 FVP_Base_Cortex-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-5 FVP_Base_Cortex-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A35x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A35x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A35x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A35x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A35x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A35x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-5 FVP_Base_Cortex-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A35x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A35x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A35x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A35x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A35x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-5 FVP_Base_Cortex-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A35x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A35x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A35x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A35x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A35x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A35x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A35x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-5 FVP_Base_Cortex-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A35x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A35x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A35x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A35x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A35x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A35x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-5 FVP_Base_Cortex-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A35x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A35x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A35x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A35x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A35x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A35x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A35x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A35x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A35x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A35x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A35x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A35x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A35x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A35x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A35x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-5 FVP_Base_Cortex-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A35x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x1.cluster0	Cluster_ARM_Cortex-A35	ARM Cortex-A35 Cluster CT model.
FVP_Base_Cortex_A35x1.cluster0.cpu0	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A35x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A35x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A35x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A35x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A35x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A35x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.6 FVP_Base_Cortex-A35x2

FVP_Base_Cortex-A35x2 contains the following instances:

Table 3-6 FVP_Base_Cortex-A35x2 instances

Name	Type	Description
FVP_Base_Cortex_A35x2	FVP_Base_Cortex_A35x2	Base Platform Compute Subsystem for ARMCortexA35x2CT.
FVP_Base_Cortex_A35x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A35x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A35x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A35x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A35x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-6 FVP_Base_Cortex-A35x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-6 FVP_Base_Cortex-A35x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A35x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A35x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A35x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A35x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A35x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A35x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-6 FVP_Base_Cortex-A35x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A35x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A35x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A35x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A35x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A35x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-6 FVP_Base_Cortex-A35x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A35x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A35x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A35x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A35x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A35x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A35x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A35x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-6 FVP_Base_Cortex-A35x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A35x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A35x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A35x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A35x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A35x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A35x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-6 FVP_Base_Cortex-A35x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A35x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A35x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A35x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A35x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A35x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A35x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A35x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A35x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A35x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A35x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A35x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A35x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A35x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A35x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A35x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-6 FVP_Base_Cortex-A35x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A35x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x2.cluster0	Cluster_ARM_Cortex-A35	ARM Cortex-A35 Cluster CT model.
FVP_Base_Cortex_A35x2.cluster0.cpu0	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A35x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A35x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x2.cluster0.cpu1	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A35x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A35x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x2.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A35x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A35x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A35x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A35x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.7 FVP_Base_Cortex-A35x4

FVP_Base_Cortex-A35x4 contains the following instances:

Table 3-7 FVP_Base_Cortex-A35x4 instances

Name	Type	Description
FVP_Base_Cortex_A35x4	FVP_Base_Cortex_A35x4	Base Platform Compute Subsystem for ARMCortexA35x4CT.
FVP_Base_Cortex_A35x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A35x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A35x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A35x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A35x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A35x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A35x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A35x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A35x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A35x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A35x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A35x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A35x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A35x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A35x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A35x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A35x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A35x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A35x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A35x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A35x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A35x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A35x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A35x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A35x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A35x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A35x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A35x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A35x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A35x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A35x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A35x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A35x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A35x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A35x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A35x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A35x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A35x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A35x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A35x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A35x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A35x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A35x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A35x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A35x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A35x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A35x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A35x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A35x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A35x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A35x4.cluster0	Cluster_ARM_Cortex-A35	ARM Cortex-A35 Cluster CT model.
FVP_Base_Cortex_A35x4.cluster0.cpu0	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A35x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A35x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0.cpu1	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A35x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A35x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0.cpu2	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A35x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A35x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0.cpu3	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A35x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A35x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-7 FVP_Base_Cortex-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A35x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A35x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A35x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A35x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A35x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A35x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.8 FVP_Base_Cortex-A53x1

FVP_Base_Cortex-A53x1 contains the following instances:

Table 3-8 FVP_Base_Cortex-A53x1 instances

Name	Type	Description
FVP_Base_Cortex_A53x1	FVP_Base_Cortex_A53x1	Base Platform Compute Subsystem for ARMCortexA53x1CT.
FVP_Base_Cortex_A53x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A53x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A53x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A53x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A53x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-8 FVP_Base_Cortex-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-8 FVP_Base_Cortex-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A53x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A53x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A53x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A53x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A53x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A53x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-8 FVP_Base_Cortex-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A53x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A53x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A53x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A53x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A53x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-8 FVP_Base_Cortex-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A53x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A53x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A53x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A53x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A53x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A53x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-8 FVP_Base_Cortex-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A53x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A53x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A53x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A53x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A53x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A53x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-8 FVP_Base_Cortex-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A53x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A53x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A53x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A53x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A53x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A53x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A53x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A53x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A53x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A53x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A53x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A53x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A53x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A53x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A53x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-8 FVP_Base_Cortex-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A53x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x1.cluster0	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A53x1.cluster0.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A53x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A53x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A53x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A53x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A53x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A53x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.9 FVP_Base_Cortex-A53x2

FVP_Base_Cortex-A53x2 contains the following instances:

Table 3-9 FVP_Base_Cortex-A53x2 instances

Name	Type	Description
FVP_Base_Cortex_A53x2	FVP_Base_Cortex_A53x2	Base Platform Compute Subsystem for ARMCortexA53x2CT.
FVP_Base_Cortex_A53x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A53x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A53x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A53x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A53x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-9 FVP_Base_Cortex-A53x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-9 FVP_Base_Cortex-A53x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A53x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A53x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A53x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A53x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A53x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A53x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-9 FVP_Base_Cortex-A53x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A53x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A53x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A53x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A53x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A53x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-9 FVP_Base_Cortex-A53x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A53x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A53x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A53x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A53x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A53x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A53x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A53x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-9 FVP_Base_Cortex-A53x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A53x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A53x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A53x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A53x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A53x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A53x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-9 FVP_Base_Cortex-A53x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A53x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A53x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A53x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A53x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A53x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A53x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A53x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A53x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A53x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A53x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A53x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A53x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A53x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A53x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A53x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-9 FVP_Base_Cortex-A53x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A53x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x2.cluster0	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A53x2.cluster0.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A53x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A53x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x2.cluster0.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A53x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A53x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x2.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A53x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A53x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A53x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A53x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.10 FVP_Base_Cortex-A53x4

FVP_Base_Cortex-A53x4 contains the following instances:

Table 3-10 FVP_Base_Cortex-A53x4 instances

Name	Type	Description
FVP_Base_Cortex_A53x4	FVP_Base_Cortex_A53x4	Base Platform Compute Subsystem for ARMCortexA53x4CT.
FVP_Base_Cortex_A53x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A53x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A53x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A53x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A53x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A53x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A53x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A53x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A53x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A53x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A53x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A53x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A53x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A53x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A53x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A53x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A53x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A53x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A53x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A53x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A53x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A53x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A53x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A53x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A53x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A53x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A53x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A53x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A53x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A53x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A53x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A53x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A53x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A53x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A53x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A53x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A53x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A53x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A53x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A53x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A53x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A53x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A53x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A53x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A53x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A53x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A53x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A53x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A53x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A53x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A53x4.cluster0	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A53x4.cluster0.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A53x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A53x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A53x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A53x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0.cpu2	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A53x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A53x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0.cpu3	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A53x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A53x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-10 FVP_Base_Cortex-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A53x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A53x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A53x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A53x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A53x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A53x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.11 FVP_Base_Cortex-A55

FVP_Base_Cortex-A55 contains the following instances:

Table 3-11 FVP_Base_Cortex-A55 instances

Name	Type	Description
FVP_Base_Cortex_A55	FVP_Base_Cortex_A55	Base Platform Compute Subsystem for ARMCortexA55CT.
FVP_Base_Cortex_A55.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A55.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55.bp.pl111_clcd.pl111x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A55.bp.pl111_clcd.pl111x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55.bp.pl111_clcd.pl111x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55.bp.pl111_clcd.pl111x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55.bp.pl111_clcd.pl111x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.cluster0	Cluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu2	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu3	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu4	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu4.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu4.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu4.l1icache	*PVCache*	PV Cache.

Table 3-11 FVP_Base_Cortex-A55 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55.cluster0.cpu4.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu5	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu5.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu5.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu5.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu5.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu6	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu6.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu6.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu6.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu6.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu7	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55.cluster0.cpu7.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55.cluster0.cpu7.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu7.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0.cpu7.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.12 FVP_Base_Cortex-A55+Cortex-A76

FVP_Base_Cortex-A55+Cortex-A76 contains the following instances:

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76	FVP_Base_Cortex_A55_Cortex_A76	Base Platform Compute Subsystem for ARMCortexA55CT_CortexA76CT.
FVP_Base_Cortex_A55_Cortex_A76.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55_Cortex_A76.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55_Cortex_A76.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A55_Cortex_A76.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55_Cortex_A76.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55_Cortex_A76.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55_Cortex_A76.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55_Cortex_A76.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55_Cortex_A76.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55_Cortex_A76.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55_Cortex_A76.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55_Cortex_A76.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55_Cortex_A76.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55_Cortex_A76.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55_Cortex_A76.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55_Cortex_A76.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55_Cortex_A76.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55_Cortex_A76.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55_Cortex_A76.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55_Cortex_A76.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55_Cortex_A76.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55_Cortex_A76.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55_Cortex_A76.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55_Cortex_A76.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55_Cortex_A76.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55_Cortex_A76.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55_Cortex_A76.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55_Cortex_A76.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55_Cortex_A76.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55_Cortex_A76.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55_Cortex_A76.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55_Cortex_A76.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55_Cortex_A76.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55_Cortex_A76.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55_Cortex_A76.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55_Cortex_A76.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55_Cortex_A76.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55_Cortex_A76.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55_Cortex_A76.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55_Cortex_A76.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55_Cortex_A76.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55_Cortex_A76.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55_Cortex_A76.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55_Cortex_A76.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55_Cortex_A76.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55_Cortex_A76.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55_Cortex_A76.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55_Cortex_A76.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55_Cortex_A76.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55_Cortex_A76.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55_Cortex_A76.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-12 FVP_Base_Cortex-A55+Cortex-A76 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55_Cortex_A76.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55_Cortex_A76.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55_Cortex_A76.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55_Cortex_A76.cluster0	Cluster_ARM_Cortex-A55_Cortex-A76	ARM Cortex-A55_Cortex-A76 Cluster CT model.
FVP_Base_Cortex_A55_Cortex_A76.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55_Cortex_A76.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55_Cortex_A76.cluster0.subcluster0	Subcluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55_Cortex_A76.cluster0.subcluster1	Subcluster_ARM_Cortex-A76	ARM Cortex-A76 Cluster CT model.
FVP_Base_Cortex_A55_Cortex_A76.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55_Cortex_A76.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55_Cortex_A76.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55_Cortex_A76.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55_Cortex_A76.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.13 FVP_Base_Cortex-A55x1

FVP_Base_Cortex-A55x1 contains the following instances:

Table 3-13 FVP_Base_Cortex-A55x1 instances

Name	Type	Description
FVP_Base_Cortex_A55x1	FVP_Base_Cortex_A55x1	Base Platform Compute Subsystem for ARMCortexA55x1CT.
FVP_Base_Cortex_A55x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-13 FVP_Base_Cortex-A55x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-13 FVP_Base_Cortex-A55x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A55x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-13 FVP_Base_Cortex-A55x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-13 FVP_Base_Cortex-A55x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A55x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-13 FVP_Base_Cortex-A55x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-13 FVP_Base_Cortex-A55x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-13 FVP_Base_Cortex-A55x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1.cluster0	Cluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x1.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.14 FVP_Base_Cortex-A55x1+Cortex-A75x1

FVP_Base_Cortex-A55x1+Cortex-A75x1 contains the following instances:

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1	FVP_Base_Cortex_A55x1_Cortex_A75x1	Base Platform Compute Subsystem for ARMCortexA55x1CT_CortexA75x1CT.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-14 FVP_Base_Cortex-A55x1+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x1_Cortex_A75x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x1_Cortex_A75x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x1_Cortex_A75x1.cluster0	Cluster_ARM_Cortex-A55_Cortex-A75	ARM Cortex-A55_Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.cluster0.cpu1	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.cluster0.subcluster0	Subcluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.cluster0.subcluster1	Subcluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x1_Cortex_A75x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x1_Cortex_A75x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x1_Cortex_A75x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55x1_Cortex_A75x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x1_Cortex_A75x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.15 FVP_Base_Cortex-A55x2

FVP_Base_Cortex-A55x2 contains the following instances:

Table 3-15 FVP_Base_Cortex-A55x2 instances

Name	Type	Description
FVP_Base_Cortex_A55x2	FVP_Base_Cortex_A55x2	Base Platform Compute Subsystem for ARMCortexA55x2CT.
FVP_Base_Cortex_A55x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-15 FVP_Base_Cortex-A55x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-15 FVP_Base_Cortex-A55x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A55x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-15 FVP_Base_Cortex-A55x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-15 FVP_Base_Cortex-A55x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A55x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-15 FVP_Base_Cortex-A55x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-15 FVP_Base_Cortex-A55x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-15 FVP_Base_Cortex-A55x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2.cluster0	Cluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x2.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x2.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.16 FVP_Base_Cortex-A55x2+Cortex-A75x2

FVP_Base_Cortex-A55x2+Cortex-A75x2 contains the following instances:

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2	FVP_Base_Cortex_A55x2_Cortex_A75x2	Base Platform Compute Subsystem for ARMCortexA55x2CT_CortexA75x2CT.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-16 FVP_Base_Cortex-A55x2+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x2_Cortex_A75x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0	Cluster_ARM_Cortex-A55_Cortex-A75	ARM Cortex-A55_Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0.cpu2	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0.cpu3	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0.subcluster0	Subcluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0.subcluster1	Subcluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x2_Cortex_A75x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x2_Cortex_A75x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x2_Cortex_A75x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55x2_Cortex_A75x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x2_Cortex_A75x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.17 FVP_Base_Cortex-A55x4

FVP_Base_Cortex-A55x4 contains the following instances:

Table 3-17 FVP_Base_Cortex-A55x4 instances

Name	Type	Description
FVP_Base_Cortex_A55x4	FVP_Base_Cortex_A55x4	Base Platform Compute Subsystem for ARMCortexA55x4CT.
FVP_Base_Cortex_A55x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A55x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A55x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4.cluster0	Cluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x4.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu2	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.

Table 3-17 FVP_Base_Cortex-A55x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu3	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A55x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A55x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.18 FVP_Base_Cortex-A55x4+Cortex-A75x1

FVP_Base_Cortex-A55x4+Cortex-A75x1 contains the following instances:

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1	FVP_Base_Cortex_A55x4_Cortex_A75x1	Base Platform Compute Subsystem for ARMCortexA55x4CT_CortexA75x1CT.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-18 FVP_Base_Cortex-A55x4+Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x4_Cortex_A75x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0	Cluster_ARM_Cortex-A55_Cortex-A75	ARM Cortex-A55_Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0.cpu2	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0.cpu3	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0.cpu4	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0.subcluster0	Subcluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0.subcluster1	Subcluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x4_Cortex_A75x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55x4_Cortex_A75x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x4_Cortex_A75x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.19 FVP_Base_Cortex-A55x4+Cortex-A75x2

FVP_Base_Cortex-A55x4+Cortex-A75x2 contains the following instances:

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2	FVP_Base_Cortex_A55x4_Cortex_A75x2	Base Platform Compute Subsystem for ARMCortexA55x4CT_CortexA75x2CT.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x4_Cortex_A75x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0	Cluster_ARM_Cortex-A55_Cortex-A75	ARM Cortex-A55_Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.cpu2	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.cpu3	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.cpu4	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.cpu5	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.subcluster0	Subcluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0.subcluster1	Subcluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x4_Cortex_A75x2.elfloader	*ElfLoader*	ELF loader component.

Table 3-19 FVP_Base_Cortex-A55x4+Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x4_Cortex_A75x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.20 FVP_Base_Cortex-A55x4+Cortex-A75x4

FVP_Base_Cortex-A55x4+Cortex-A75x4 contains the following instances:

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4	FVP_Base_Cortex_A55x4_Cortex_A75x4	Base Platform Compute Subsystem for ARMCortexA55x4CT_CortexA75x4CT.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x4_Cortex_A75x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0	Cluster_ARM_Cortex-A55_Cortex-A75	ARM Cortex-A55_Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu2	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu3	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu4	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu5	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu6	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.cpu7	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.subcluster0	Subcluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0.subcluster1	Subcluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A75x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A75x4.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-20 FVP_Base_Cortex-A55x4+Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A75x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A55x4_Cortex_A75x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x4_Cortex_A75x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.21 FVP_Base_Cortex-A55x4+Cortex-A76x2

FVP_Base_Cortex-A55x4+Cortex-A76x2 contains the following instances:

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2	FVP_Base_Cortex_A55x4_Cortex_A76x2	Base Platform Compute Subsystem for ARMCortexA55x4CT_CortexA76x2CT.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A55x4_Cortex_A76x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0	Cluster_ARM_Cortex-A55_Cortex-A76	ARM Cortex-A55_Cortex-A76 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.cpu0	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.cpu1	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.cpu2	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.cpu3	ARM_Cortex-A55	ARM Cortex-A55 CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.cpu4	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.cpu5	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.subcluster0	Subcluster_ARM_Cortex-A55	ARM Cortex-A55 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0.subcluster1	Subcluster_ARM_Cortex-A76	ARM Cortex-A76 Cluster CT model.
FVP_Base_Cortex_A55x4_Cortex_A76x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A55x4_Cortex_A76x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A55x4_Cortex_A76x2.elfloader	*ElfLoader*	ELF loader component.

Table 3-21 FVP_Base_Cortex-A55x4+Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A55x4_Cortex_A76x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A55x4_Cortex_A76x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.22 FVP_Base_Cortex-A57x1

FVP_Base_Cortex-A57x1 contains the following instances:

Table 3-22 FVP_Base_Cortex-A57x1 instances

Name	Type	Description
FVP_Base_Cortex_A57x1	FVP_Base_Cortex_A57x1	Base Platform Compute Subsystem for ARMCortexA57x1CT.
FVP_Base_Cortex_A57x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A57x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-22 FVP_Base_Cortex-A57x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-22 FVP_Base_Cortex-A57x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-22 FVP_Base_Cortex-A57x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-22 FVP_Base_Cortex-A57x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A57x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-22 FVP_Base_Cortex-A57x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-22 FVP_Base_Cortex-A57x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A57x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-22 FVP_Base_Cortex-A57x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A57x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x1.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A57x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.23 FVP_Base_Cortex-A57x1-A35x1

FVP_Base_Cortex-A57x1-A35x1 contains the following instances:

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1	FVP_Base_Cortex_A57x1_A35x1	Base Platform Compute Subsystem for ARMCortexA57x1CT and ARMCortexA35x1CT.
FVP_Base_Cortex_A57x1_A35x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A35x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A35x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A57x1_A35x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x1_A35x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x1_A35x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x1_A35x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A35x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A35x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A35x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A35x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x1_A35x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x1_A35x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1_A35x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x1_A35x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x1_A35x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A35x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x1_A35x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x1_A35x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A35x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1_A35x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x1_A35x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x1_A35x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x1_A35x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x1_A35x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x1_A35x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1_A35x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1_A35x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x1_A35x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x1_A35x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A35x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x1_A35x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x1_A35x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x1_A35x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x1_A35x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x1_A35x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A35x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A35x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x1_A35x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x1_A35x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x1_A35x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A35x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A35x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A35x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A35x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x1_A35x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x1_A35x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x1_A35x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x1_A35x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x1_A35x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x1_A35x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x1_A35x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A35x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x1_A35x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A35x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x1_A35x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A35x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x1_A35x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x1_A35x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A35x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A35x1.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x1_A35x1.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x1_A35x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x1_A35x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A35x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A35x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A35x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A35x1.cluster1	Cluster_ARM_Cortex-A35	ARM Cortex-A35 Cluster CT model.
FVP_Base_Cortex_A57x1_A35x1.cluster1.cpu0	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x1_A35x1.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x1_A35x1.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A35x1.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A35x1.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A35x1.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A35x1.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-23 FVP_Base_Cortex-A57x1-A35x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A35x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x1_A35x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x1_A35x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.24 FVP_Base_Cortex-A57x1-A53x1

FVP_Base_Cortex-A57x1-A53x1 contains the following instances:

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1	FVP_Base_Cortex_A57x1_A53x1	Base Platform Compute Subsystem for ARMCortexA57x1CT and ARMCortexA53x1CT.
FVP_Base_Cortex_A57x1_A53x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A53x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x1_A53x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A57x1_A53x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x1_A53x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x1_A53x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x1_A53x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A53x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A53x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A53x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A53x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x1_A53x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x1_A53x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1_A53x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x1_A53x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x1_A53x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A53x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x1_A53x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x1_A53x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A53x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x1_A53x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x1_A53x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x1_A53x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x1_A53x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x1_A53x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x1_A53x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x1_A53x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A53x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x1_A53x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x1_A53x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x1_A53x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x1_A53x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x1_A53x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x1_A53x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x1_A53x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x1_A53x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x1_A53x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x1_A53x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A53x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A53x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A53x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x1_A53x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x1_A53x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x1_A53x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x1_A53x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x1_A53x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x1_A53x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x1_A53x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x1_A53x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A53x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x1_A53x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A53x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x1_A53x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A53x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x1_A53x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x1_A53x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x1_A53x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x1_A53x1.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x1_A53x1.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x1_A53x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x1_A53x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A53x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A53x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A53x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A53x1.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A57x1_A53x1.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x1_A53x1.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x1_A53x1.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A53x1.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A53x1.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x1_A53x1.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A57x1_A53x1.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-24 FVP_Base_Cortex-A57x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x1_A53x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x1_A53x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x1_A53x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.25 FVP_Base_Cortex-A57x2

FVP_Base_Cortex-A57x2 contains the following instances:

Table 3-25 FVP_Base_Cortex-A57x2 instances

Name	Type	Description
FVP_Base_Cortex_A57x2	FVP_Base_Cortex_A57x2	Base Platform Compute Subsystem for ARMCortexA57x2CT.
FVP_Base_Cortex_A57x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A57x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-25 FVP_Base_Cortex-A57x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-25 FVP_Base_Cortex-A57x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-25 FVP_Base_Cortex-A57x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-25 FVP_Base_Cortex-A57x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A57x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-25 FVP_Base_Cortex-A57x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-25 FVP_Base_Cortex-A57x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A57x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-25 FVP_Base_Cortex-A57x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A57x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x2.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2.cluster0.cpu1	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A57x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.26 FVP_Base_Cortex-A57x2-A35x4

FVP_Base_Cortex-A57x2-A35x4 contains the following instances:

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4	FVP_Base_Cortex_A57x2_A35x4	Base Platform Compute Subsystem for ARMCortexA57x2CT and ARMCortexA35x4CT.
FVP_Base_Cortex_A57x2_A35x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A35x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A35x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A57x2_A35x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x2_A35x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x2_A35x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x2_A35x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A35x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A35x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A35x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A35x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x2_A35x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x2_A35x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2_A35x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x2_A35x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x2_A35x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A35x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x2_A35x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x2_A35x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A35x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2_A35x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x2_A35x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x2_A35x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x2_A35x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x2_A35x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x2_A35x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x2_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x2_A35x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A35x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x2_A35x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x2_A35x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x2_A35x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x2_A35x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x2_A35x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A35x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A35x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x2_A35x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x2_A35x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x2_A35x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A35x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A35x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A35x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A35x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x2_A35x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x2_A35x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x2_A35x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x2_A35x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x2_A35x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x2_A35x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x2_A35x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A35x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x2_A35x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A35x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x2_A35x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A35x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x2_A35x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x2_A35x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A35x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A35x4.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu1	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A35x4.cluster1	Cluster_ARM_Cortex-A35	ARM Cortex-A35 Cluster CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu0	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.

Table 3-26 FVP_Base_Cortex-A57x2-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu1	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu2	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu3	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A35x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A35x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A57x2_A35x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x2_A35x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x2_A35x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.27 FVP_Base_Cortex-A57x2-A53x4

FVP_Base_Cortex-A57x2-A53x4 contains the following instances:

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4	FVP_Base_Cortex_A57x2_A53x4	Base Platform Compute Subsystem for ARMCortexA57x2CT and ARMCortexA53x4CT.
FVP_Base_Cortex_A57x2_A53x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A53x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x2_A53x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A57x2_A53x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x2_A53x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x2_A53x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x2_A53x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A53x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A53x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A53x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A53x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x2_A53x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x2_A53x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2_A53x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x2_A53x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x2_A53x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A53x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x2_A53x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x2_A53x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A53x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x2_A53x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x2_A53x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x2_A53x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x2_A53x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x2_A53x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x2_A53x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x2_A53x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A53x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x2_A53x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x2_A53x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x2_A53x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x2_A53x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x2_A53x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x2_A53x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x2_A53x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x2_A53x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x2_A53x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x2_A53x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A53x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A53x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A53x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x2_A53x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x2_A53x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x2_A53x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x2_A53x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x2_A53x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x2_A53x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x2_A53x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x2_A53x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A53x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x2_A53x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A53x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x2_A53x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A53x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x2_A53x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x2_A53x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x2_A53x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x2_A53x4.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu1	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A53x4.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.

Table 3-27 FVP_Base_Cortex-A57x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu2	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu3	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x2_A53x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A57x2_A53x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A57x2_A53x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x2_A53x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x2_A53x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.28 FVP_Base_Cortex-A57x4

FVP_Base_Cortex-A57x4 contains the following instances:

Table 3-28 FVP_Base_Cortex-A57x4 instances

Name	Type	Description
FVP_Base_Cortex_A57x4	FVP_Base_Cortex_A57x4	Base Platform Compute Subsystem for ARMCortexA57x4CT.
FVP_Base_Cortex_A57x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A57x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A57x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A57x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A57x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x4.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0.cpu1	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0.cpu2	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0.cpu3	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-28 FVP_Base_Cortex-A57x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A57x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.29 FVP_Base_Cortex-A57x4-A35x4

FVP_Base_Cortex-A57x4-A35x4 contains the following instances:

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4	FVP_Base_Cortex_A57x4_A35x4	Base Platform Compute Subsystem for ARMCortexA57x4CT and ARMCortexA35x4CT.
FVP_Base_Cortex_A57x4_A35x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A35x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A35x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A57x4_A35x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x4_A35x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x4_A35x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x4_A35x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A35x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A35x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A35x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A35x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x4_A35x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x4_A35x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4_A35x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x4_A35x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x4_A35x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A35x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x4_A35x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x4_A35x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A35x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4_A35x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x4_A35x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x4_A35x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x4_A35x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x4_A35x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x4_A35x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x4_A35x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x4_A35x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A35x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x4_A35x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x4_A35x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x4_A35x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x4_A35x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x4_A35x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A35x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A35x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x4_A35x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x4_A35x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x4_A35x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A35x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A35x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A35x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A35x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x4_A35x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x4_A35x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x4_A35x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x4_A35x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x4_A35x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x4_A35x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x4_A35x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A35x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x4_A35x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A35x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x4_A35x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A35x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x4_A35x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x4_A35x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A35x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A35x4.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu1	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu2	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu3	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A35x4.cluster1	Cluster_ARM_Cortex-A35	ARM Cortex-A35 Cluster CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu0	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu1	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu2	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu3	ARM_Cortex-A35	ARM Cortex-A35 CT model.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.

Table 3-29 FVP_Base_Cortex-A57x4-A35x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A35x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A35x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A35x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A57x4_A35x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x4_A35x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x4_A35x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.30 FVP_Base_Cortex-A57x4-A53x4

FVP_Base_Cortex-A57x4-A53x4 contains the following instances:

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4	FVP_Base_Cortex_A57x4_A53x4	Base Platform Compute Subsystem for ARMCortexA57x4CT and ARMCortexA53x4CT.
FVP_Base_Cortex_A57x4_A53x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A53x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A57x4_A53x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A57x4_A53x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x4_A53x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A57x4_A53x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A57x4_A53x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A53x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A53x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A53x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A53x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A57x4_A53x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A57x4_A53x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4_A53x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x4_A53x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x4_A53x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A53x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A57x4_A53x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A57x4_A53x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A53x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A57x4_A53x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A57x4_A53x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A57x4_A53x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x4_A53x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A57x4_A53x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A57x4_A53x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A57x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A57x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A57x4_A53x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A53x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A57x4_A53x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x4_A53x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A57x4_A53x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A57x4_A53x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A57x4_A53x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A57x4_A53x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A57x4_A53x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A57x4_A53x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x4_A53x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A57x4_A53x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A53x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A53x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A53x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A57x4_A53x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A57x4_A53x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A57x4_A53x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A57x4_A53x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A57x4_A53x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A57x4_A53x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A57x4_A53x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A57x4_A53x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A53x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A57x4_A53x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A53x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A57x4_A53x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A53x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A57x4_A53x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A57x4_A53x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A57x4_A53x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A57x4_A53x4.cluster0	Cluster_ARM_Cortex-A57	ARM Cortex-A57 Cluster CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu0	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu1	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu2	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu3	ARM_Cortex-A57	ARM Cortex-A57 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A53x4.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu2	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu3	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.

Table 3-30 FVP_Base_Cortex-A57x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A57x4_A53x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A57x4_A53x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A57x4_A53x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A57x4_A53x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A57x4_A53x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A57x4_A53x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.31 FVP_Base_Cortex-A65AEx2

FVP_Base_Cortex-A65AEx2 contains the following instances:

Table 3-31 FVP_Base_Cortex-A65AEx2 instances

Name	Type	Description
FVP_Base_Cortex_A65AEx2	FVP_Base_Cortex_A65AEx2	Base Platform Compute Subsystem for ARMCortexA65AEx2CT.
FVP_Base_Cortex_A65AEx2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65AEx2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A65AEx2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65AEx2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65AEx2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A65AEx2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65AEx2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65AEx2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65AEx2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65AEx2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65AEx2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65AEx2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65AEx2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65AEx2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.securefla sh	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.bp.securefla shloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2.bp.smsc_91c 111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65AEx2.bp.sp805_w dog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx2.bp.sp810_sy sctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65AEx2.bp.sp810_sy sctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.sp810_sy sctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.sp810_sy sctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.sp810_sy sctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.bp.terminal_ 0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx2.bp.terminal_ 1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx2.bp.terminal_ 2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx2.bp.terminal_ 3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx2.bp.trusted_k ey_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65AEx2.bp.trusted_n v_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65AEx2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65AEx2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65AEx2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65AEx2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65AEx2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65AEx2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65AEx2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65AEx2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65AEx2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65AEx2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-31 FVP_Base_Cortex-A65AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2.cluster0	Cluster_ARM_Cortex-A65AE	ARM Cortex-A65AE Cluster CT model.
FVP_Base_Cortex_A65AEx2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx2.cluster0.cpu0.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2.cluster0.cpu0.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx2.cluster0.cpu1.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2.cluster0.cpu1.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A65AEx2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A65AEx2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65AEx2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2

FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 contains the following instances:

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2	FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2	Base Platform Compute Subsystem for ARMCortexA65AEx2CT_CortexA76AEx2CT.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0	Cluster_ARM_Cortex-A65AE_Cortex-A76AE	ARM Cortex-A65AE_Cortex-A76AE Cluster CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.cpu0.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.cpu0.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.cpu1.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.cpu1.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.cpu2	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.cpu3	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.subcluster0	Subcluster_ARM_Cortex-A65AE	ARM Cortex-A65AE Cluster CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0.subcluster1	Subcluster_ARM_Cortex-A76AE	ARM Cortex-A76AE Cluster CT model.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.elfloader	*ElfLoader*	ELF loader component.

Table 3-32 FVP_Base_Cortex-A65AEx2+Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65AEx2_Cortex_A76AEx2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.33 FVP_Base_Cortex-A65AEx4

FVP_Base_Cortex-A65AEx4 contains the following instances:

Table 3-33 FVP_Base_Cortex-A65AEx4 instances

Name	Type	Description
FVP_Base_Cortex_A65AEx4	FVP_Base_Cortex_A65AEx4	Base Platform Compute Subsystem for ARMCortexA65AEx4CT.
FVP_Base_Cortex_A65AEx4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65AEx4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A65AEx4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65AEx4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65AEx4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A65AEx4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65AEx4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65AEx4.bp.nontrusteddrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx4.bp.nontrusteddromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65AEx4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65AEx4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65AEx4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65AEx4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65AEx4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65AEx4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.securefla sh	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.bp.securefla shloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4.bp.smsc_91c 111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65AEx4.bp.sp805_w dog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx4.bp.sp810_sy sctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65AEx4.bp.sp810_sy sctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.sp810_sy sctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.sp810_sy sctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.sp810_sy sctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.bp.terminal_ 0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx4.bp.terminal_ 1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx4.bp.terminal_ 2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx4.bp.terminal_ 3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx4.bp.trusted_k ey_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65AEx4.bp.trusted_n v_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65AEx4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65AEx4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65AEx4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65AEx4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65AEx4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65AEx4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65AEx4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65AEx4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65AEx4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65AEx4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.cluster0	Cluster_ARM_Cortex-A65AE	ARM Cortex-A65AE Cluster CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu0.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu0.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu1.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu1.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu2.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu2.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-33 FVP_Base_Cortex-A65AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx4.cluster0.cpu3.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0.cpu3.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A65AEx4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A65AEx4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65AEx4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4

FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 contains the following instances:

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4	FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4	Base Platform Compute Subsystem for ARMCortexA65AEx4CT_CortexA76AEx4CT.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0	Cluster_ARM_Cortex-A65AE_Cortex-A76AE	ARM Cortex-A65AE_Cortex-A76AE Cluster CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu0.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu0.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu1.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu1.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu2.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu2.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu3.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu3.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu4	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu5	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu6	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.cpu7	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.

Table 3-34 FVP_Base_Cortex-A65AEx4+Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.subcluster0	Subcluster_ARM_Cortex-A65AE	ARM Cortex-A65AE Cluster CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0.subcluster1	Subcluster_ARM_Cortex-A76AE	ARM Cortex-A76AE Cluster CT model.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65AEx4_Cortex_A76AEx4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.35 FVP_Base_Cortex-A65AEx8

FVP_Base_Cortex-A65AEx8 contains the following instances:

Table 3-35 FVP_Base_Cortex-A65AEx8 instances

Name	Type	Description
FVP_Base_Cortex_A65AEx8	FVP_Base_Cortex_A65AEx8	Base Platform Compute Subsystem for ARMCortexA65AEx8CT.
FVP_Base_Cortex_A65AEx8.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65AEx8.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx8.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx8.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx8.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65AEx8.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx8.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65AEx8.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A65AEx8.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx8.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65AEx8.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65AEx8.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx8.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx8.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx8.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx8.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65AEx8.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A65AEx8.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx8.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx8.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx8.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx8.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65AEx8.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65AEx8.bp.nontrusteddrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65AEx8.bp.nontrusteddromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx8.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx8.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx8.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65AEx8.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65AEx8.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65AEx8.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx8.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65AEx8.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65AEx8.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx8.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65AEx8.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65AEx8.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65AEx8.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx8.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65AEx8.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx8.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65AEx8.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65AEx8.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65AEx8.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.securefla sh	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.bp.securefla shloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65AEx8.bp.smsc_91c 111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65AEx8.bp.sp805_w dog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx8.bp.sp810_sy sctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65AEx8.bp.sp810_sy sctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.sp810_sy sctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.sp810_sy sctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.sp810_sy sctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.bp.terminal_ 0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx8.bp.terminal_ 1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx8.bp.terminal_ 2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx8.bp.terminal_ 3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65AEx8.bp.trusted_k ey_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65AEx8.bp.trusted_n v_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65AEx8.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65AEx8.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65AEx8.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65AEx8.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65AEx8.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65AEx8.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx8.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65AEx8.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx8.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65AEx8.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx8.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65AEx8.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65AEx8.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65AEx8.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65AEx8.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65AEx8.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.cluster0	Cluster_ARM_Cortex-A65AE	ARM Cortex-A65AE Cluster CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu0.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu0.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu1.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu1.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu2.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu2.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu3.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu3.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu4.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu4.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu4.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu4.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu4.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu4.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu5.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu5.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu5.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu5.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu5.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu5.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu6.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu6.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu6.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu6.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu6.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.

Table 3-35 FVP_Base_Cortex-A65AEx8 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65AEx8.cluster0.cpu6.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu7.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65AEx8.cluster0.cpu7.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu7.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu7.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65AEx8.cluster0.cpu7.thread0	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0.cpu7.thread1	ARM_Cortex-A65AE	ARM Cortex-A65AE CT model.
FVP_Base_Cortex_A65AEx8.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65AEx8.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A65AEx8.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A65AEx8.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65AEx8.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.36 FVP_Base_Cortex-A65x1

FVP_Base_Cortex-A65x1 contains the following instances:

Table 3-36 FVP_Base_Cortex-A65x1 instances

Name	Type	Description
FVP_Base_Cortex_A65x1	FVP_Base_Cortex_A65x1	Base Platform Compute Subsystem for ARMCortexA65x1CT.
FVP_Base_Cortex_A65x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A65x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-36 FVP_Base_Cortex-A65x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-36 FVP_Base_Cortex-A65x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A65x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-36 FVP_Base_Cortex-A65x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-36 FVP_Base_Cortex-A65x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A65x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-36 FVP_Base_Cortex-A65x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-36 FVP_Base_Cortex-A65x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A65x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-36 FVP_Base_Cortex-A65x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x1.cluster0	Cluster_ARM_Cortex-A65	ARM Cortex-A65 Cluster CT model.
FVP_Base_Cortex_A65x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x1.cluster0.cpu0.thread0	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x1.cluster0.cpu0.thread1	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A65x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A65x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.37 FVP_Base_Cortex-A65x2

FVP_Base_Cortex-A65x2 contains the following instances:

Table 3-37 FVP_Base_Cortex-A65x2 instances

Name	Type	Description
FVP_Base_Cortex_A65x2	FVP_Base_Cortex_A65x2	Base Platform Compute Subsystem for ARMCortexA65x2CT.
FVP_Base_Cortex_A65x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A65x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A65x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A65x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A65x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x2.cluster0	Cluster_ARM_Cortex-A65	ARM Cortex-A65 Cluster CT model.
FVP_Base_Cortex_A65x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x2.cluster0.cpu0.thread0	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x2.cluster0.cpu0.thread1	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x2.cluster0.cpu1.thread0	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x2.cluster0.cpu1.thread1	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65x2.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-37 FVP_Base_Cortex-A65x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A65x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.38 FVP_Base_Cortex-A65x4

FVP_Base_Cortex-A65x4 contains the following instances:

Table 3-38 FVP_Base_Cortex-A65x4 instances

Name	Type	Description
FVP_Base_Cortex_A65x4	FVP_Base_Cortex_A65x4	Base Platform Compute Subsystem for ARMCortexA65x4CT.
FVP_Base_Cortex_A65x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A65x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A65x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A65x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A65x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A65x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A65x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A65x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A65x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A65x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A65x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A65x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A65x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A65x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A65x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A65x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A65x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A65x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A65x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A65x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A65x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A65x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A65x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A65x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A65x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A65x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A65x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A65x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A65x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A65x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A65x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A65x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A65x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A65x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A65x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A65x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A65x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A65x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A65x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A65x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A65x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A65x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A65x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A65x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A65x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A65x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A65x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A65x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A65x4.cluster0	Cluster_ARM_Cortex-A65	ARM Cortex-A65 Cluster CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu0.thread0	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu0.thread1	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu1.thread0	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu1.thread1	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-38 FVP_Base_Cortex-A65x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A65x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu2.thread0	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu2.thread1	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A65x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A65x4.cluster0.cpu3.thread0	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0.cpu3.thread1	ARM_Cortex-A65	ARM Cortex-A65 CT model.
FVP_Base_Cortex_A65x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A65x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A65x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A65x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A65x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.39 FVP_Base_Cortex-A72x1

FVP_Base_Cortex-A72x1 contains the following instances:

Table 3-39 FVP_Base_Cortex-A72x1 instances

Name	Type	Description
FVP_Base_Cortex_A72x1	FVP_Base_Cortex_A72x1	Base Platform Compute Subsystem for ARMCortexA72x1CT.
FVP_Base_Cortex_A72x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A72x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A72x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-39 FVP_Base_Cortex-A72x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-39 FVP_Base_Cortex-A72x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A72x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A72x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A72x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-39 FVP_Base_Cortex-A72x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A72x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A72x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A72x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A72x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-39 FVP_Base_Cortex-A72x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A72x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A72x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A72x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-39 FVP_Base_Cortex-A72x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A72x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A72x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A72x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A72x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-39 FVP_Base_Cortex-A72x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A72x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A72x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A72x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A72x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A72x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A72x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A72x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A72x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A72x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A72x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A72x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-39 FVP_Base_Cortex-A72x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A72x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1.cluster0	Cluster_ARM_Cortex-A72	ARM Cortex-A72 Cluster CT model.
FVP_Base_Cortex_A72x1.cluster0.cpu0	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A72x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A72x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A72x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.40 FVP_Base_Cortex-A72x1-A53x1

FVP_Base_Cortex-A72x1-A53x1 contains the following instances:

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1	FVP_Base_Cortex_A72x1_A53x1	Base Platform Compute Subsystem for ARMCortexA72x1CT and ARMCortexA53x1CT.
FVP_Base_Cortex_A72x1_A53x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1_A53x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x1_A53x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A72x1_A53x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x1_A53x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x1_A53x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A72x1_A53x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1_A53x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1_A53x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1_A53x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1_A53x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A72x1_A53x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A72x1_A53x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x1_A53x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x1_A53x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x1_A53x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1_A53x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A72x1_A53x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A72x1_A53x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1_A53x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x1_A53x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A72x1_A53x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A72x1_A53x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x1_A53x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x1_A53x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A72x1_A53x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x1_A53x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1_A53x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A72x1_A53x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x1_A53x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x1_A53x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A72x1_A53x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A72x1_A53x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x1_A53x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x1_A53x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A72x1_A53x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x1_A53x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A72x1_A53x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1_A53x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1_A53x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1_A53x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x1_A53x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A72x1_A53x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A72x1_A53x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x1_A53x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A72x1_A53x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A72x1_A53x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A72x1_A53x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A72x1_A53x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1_A53x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A72x1_A53x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1_A53x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A72x1_A53x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1_A53x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A72x1_A53x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A72x1_A53x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x1_A53x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x1_A53x1.cluster0	Cluster_ARM_Cortex-A72	ARM Cortex-A72 Cluster CT model.
FVP_Base_Cortex_A72x1_A53x1.cluster0.cpu0	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x1_A53x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x1_A53x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1_A53x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1_A53x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1_A53x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1_A53x1.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A72x1_A53x1.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x1_A53x1.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x1_A53x1.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1_A53x1.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1_A53x1.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x1_A53x1.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A72x1_A53x1.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-40 FVP_Base_Cortex-A72x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x1_A53x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A72x1_A53x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A72x1_A53x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.41 FVP_Base_Cortex-A72x2

FVP_Base_Cortex-A72x2 contains the following instances:

Table 3-41 FVP_Base_Cortex-A72x2 instances

Name	Type	Description
FVP_Base_Cortex_A72x2	FVP_Base_Cortex_A72x2	Base Platform Compute Subsystem for ARMCortexA72x2CT.
FVP_Base_Cortex_A72x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A72x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A72x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-41 FVP_Base_Cortex-A72x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-41 FVP_Base_Cortex-A72x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A72x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A72x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A72x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-41 FVP_Base_Cortex-A72x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A72x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A72x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A72x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A72x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-41 FVP_Base_Cortex-A72x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A72x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A72x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A72x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-41 FVP_Base_Cortex-A72x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A72x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A72x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A72x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A72x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-41 FVP_Base_Cortex-A72x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A72x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A72x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A72x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A72x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A72x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A72x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A72x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A72x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A72x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A72x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A72x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-41 FVP_Base_Cortex-A72x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A72x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2.cluster0	Cluster_ARM_Cortex-A72	ARM Cortex-A72 Cluster CT model.
FVP_Base_Cortex_A72x2.cluster0.cpu0	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2.cluster0.cpu1	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A72x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A72x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A72x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.42 FVP_Base_Cortex-A72x2-A53x4

FVP_Base_Cortex-A72x2-A53x4 contains the following instances:

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4	FVP_Base_Cortex_A72x2_A53x4	Base Platform Compute Subsystem for ARMCortexA72x2CT and ARMCortexA53x4CT.
FVP_Base_Cortex_A72x2_A53x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2_A53x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x2_A53x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A72x2_A53x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x2_A53x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x2_A53x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A72x2_A53x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2_A53x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2_A53x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2_A53x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2_A53x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A72x2_A53x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A72x2_A53x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x2_A53x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x2_A53x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x2_A53x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2_A53x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A72x2_A53x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A72x2_A53x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2_A53x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x2_A53x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A72x2_A53x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A72x2_A53x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x2_A53x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x2_A53x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A72x2_A53x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x2_A53x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2_A53x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A72x2_A53x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x2_A53x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x2_A53x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A72x2_A53x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A72x2_A53x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x2_A53x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x2_A53x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A72x2_A53x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x2_A53x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A72x2_A53x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2_A53x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2_A53x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2_A53x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x2_A53x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A72x2_A53x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A72x2_A53x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x2_A53x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A72x2_A53x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A72x2_A53x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A72x2_A53x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A72x2_A53x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2_A53x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A72x2_A53x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2_A53x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A72x2_A53x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2_A53x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A72x2_A53x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A72x2_A53x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x2_A53x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x2_A53x4.cluster0	Cluster_ARM_Cortex-A72	ARM Cortex-A72 Cluster CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu0	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu1	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2_A53x4.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.

Table 3-42 FVP_Base_Cortex-A72x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu2	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu3	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x2_A53x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A72x2_A53x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A72x2_A53x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A72x2_A53x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A72x2_A53x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.43 FVP_Base_Cortex-A72x4

FVP_Base_Cortex-A72x4 contains the following instances:

Table 3-43 FVP_Base_Cortex-A72x4 instances

Name	Type	Description
FVP_Base_Cortex_A72x4	FVP_Base_Cortex_A72x4	Base Platform Compute Subsystem for ARMCortexA72x4CT.
FVP_Base_Cortex_A72x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A72x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A72x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A72x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A72x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A72x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A72x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A72x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A72x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A72x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A72x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A72x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A72x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A72x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A72x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A72x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A72x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A72x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A72x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A72x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A72x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A72x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A72x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A72x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A72x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A72x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A72x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A72x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A72x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4.cluster0	Cluster_ARM_Cortex-A72	ARM Cortex-A72 Cluster CT model.
FVP_Base_Cortex_A72x4.cluster0.cpu0	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0.cpu1	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0.cpu2	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0.cpu3	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-43 FVP_Base_Cortex-A72x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A72x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A72x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A72x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.44 FVP_Base_Cortex-A72x4-A53x4

FVP_Base_Cortex-A72x4-A53x4 contains the following instances:

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4	FVP_Base_Cortex_A72x4_A53x4	Base Platform Compute Subsystem for ARMCortexA72x4CT and ARMCortexA53x4CT.
FVP_Base_Cortex_A72x4_A53x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4_A53x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A72x4_A53x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A72x4_A53x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x4_A53x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A72x4_A53x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A72x4_A53x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4_A53x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4_A53x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4_A53x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4_A53x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A72x4_A53x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A72x4_A53x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x4_A53x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x4_A53x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x4_A53x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4_A53x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A72x4_A53x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A72x4_A53x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4_A53x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A72x4_A53x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A72x4_A53x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A72x4_A53x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x4_A53x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A72x4_A53x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A72x4_A53x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A72x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A72x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A72x4_A53x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4_A53x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A72x4_A53x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x4_A53x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A72x4_A53x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A72x4_A53x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A72x4_A53x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A72x4_A53x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A72x4_A53x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A72x4_A53x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x4_A53x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A72x4_A53x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4_A53x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4_A53x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4_A53x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A72x4_A53x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A72x4_A53x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A72x4_A53x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A72x4_A53x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A72x4_A53x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A72x4_A53x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A72x4_A53x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A72x4_A53x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4_A53x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A72x4_A53x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4_A53x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A72x4_A53x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4_A53x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A72x4_A53x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A72x4_A53x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A72x4_A53x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A72x4_A53x4.cluster0	Cluster_ARM_Cortex-A72	ARM Cortex-A72 Cluster CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu0	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu1	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu2	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu3	ARM_Cortex-A72	ARM Cortex-A72 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4_A53x4.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu2	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu3	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.

Table 3-44 FVP_Base_Cortex-A72x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A72x4_A53x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A72x4_A53x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A72x4_A53x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A72x4_A53x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A72x4_A53x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A72x4_A53x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.45 FVP_Base_Cortex-A73x1

FVP_Base_Cortex-A73x1 contains the following instances:

Table 3-45 FVP_Base_Cortex-A73x1 instances

Name	Type	Description
FVP_Base_Cortex_A73x1	FVP_Base_Cortex_A73x1	Base Platform Compute Subsystem for ARMCortexA73x1CT.
FVP_Base_Cortex_A73x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A73x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A73x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-45 FVP_Base_Cortex-A73x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-45 FVP_Base_Cortex-A73x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A73x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A73x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A73x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-45 FVP_Base_Cortex-A73x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A73x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A73x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A73x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A73x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-45 FVP_Base_Cortex-A73x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A73x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A73x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A73x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-45 FVP_Base_Cortex-A73x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A73x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A73x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A73x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A73x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-45 FVP_Base_Cortex-A73x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A73x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A73x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A73x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A73x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A73x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A73x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A73x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A73x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A73x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A73x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A73x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-45 FVP_Base_Cortex-A73x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A73x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1.cluster0	Cluster_ARM_Cortex-A73	ARM Cortex-A73 Cluster CT model.
FVP_Base_Cortex_A73x1.cluster0.cpu0	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A73x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A73x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A73x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.46 FVP_Base_Cortex-A73x1-A53x1

FVP_Base_Cortex-A73x1-A53x1 contains the following instances:

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1	FVP_Base_Cortex_A73x1_A53x1	Base Platform Compute Subsystem for ARMCortexA73x1CT and ARMCortexA53x1CT.
FVP_Base_Cortex_A73x1_A53x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1_A53x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x1_A53x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A73x1_A53x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x1_A53x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x1_A53x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A73x1_A53x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1_A53x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1_A53x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1_A53x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1_A53x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A73x1_A53x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A73x1_A53x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x1_A53x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x1_A53x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x1_A53x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1_A53x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A73x1_A53x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A73x1_A53x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1_A53x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x1_A53x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A73x1_A53x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A73x1_A53x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x1_A53x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x1_A53x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A73x1_A53x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x1_A53x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x1_A53x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1_A53x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A73x1_A53x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x1_A53x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x1_A53x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A73x1_A53x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A73x1_A53x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x1_A53x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x1_A53x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A73x1_A53x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x1_A53x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A73x1_A53x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1_A53x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1_A53x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1_A53x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x1_A53x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A73x1_A53x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A73x1_A53x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x1_A53x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A73x1_A53x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A73x1_A53x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A73x1_A53x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A73x1_A53x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1_A53x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A73x1_A53x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1_A53x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A73x1_A53x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1_A53x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A73x1_A53x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A73x1_A53x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x1_A53x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x1_A53x1.cluster0	Cluster_ARM_Cortex-A73	ARM Cortex-A73 Cluster CT model.
FVP_Base_Cortex_A73x1_A53x1.cluster0.cpu0	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x1_A53x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x1_A53x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1_A53x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1_A53x1.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1_A53x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1_A53x1.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A73x1_A53x1.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x1_A53x1.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x1_A53x1.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1_A53x1.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1_A53x1.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x1_A53x1.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A73x1_A53x1.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-46 FVP_Base_Cortex-A73x1-A53x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x1_A53x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A73x1_A53x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A73x1_A53x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.47 FVP_Base_Cortex-A73x2

FVP_Base_Cortex-A73x2 contains the following instances:

Table 3-47 FVP_Base_Cortex-A73x2 instances

Name	Type	Description
FVP_Base_Cortex_A73x2	FVP_Base_Cortex_A73x2	Base Platform Compute Subsystem for ARMCortexA73x2CT.
FVP_Base_Cortex_A73x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A73x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A73x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-47 FVP_Base_Cortex-A73x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-47 FVP_Base_Cortex-A73x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A73x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A73x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A73x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-47 FVP_Base_Cortex-A73x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A73x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A73x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A73x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A73x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-47 FVP_Base_Cortex-A73x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A73x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A73x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A73x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-47 FVP_Base_Cortex-A73x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A73x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A73x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A73x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A73x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-47 FVP_Base_Cortex-A73x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A73x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A73x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A73x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A73x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A73x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A73x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A73x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A73x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A73x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A73x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A73x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-47 FVP_Base_Cortex-A73x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A73x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2.cluster0	Cluster_ARM_Cortex-A73	ARM Cortex-A73 Cluster CT model.
FVP_Base_Cortex_A73x2.cluster0.cpu0	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2.cluster0.cpu1	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A73x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A73x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A73x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.48 FVP_Base_Cortex-A73x2-A53x4

FVP_Base_Cortex-A73x2-A53x4 contains the following instances:

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4	FVP_Base_Cortex_A73x2_A53x4	Base Platform Compute Subsystem for ARMCortexA73x2CT and ARMCortexA53x4CT.
FVP_Base_Cortex_A73x2_A53x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2_A53x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x2_A53x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A73x2_A53x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x2_A53x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x2_A53x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A73x2_A53x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2_A53x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2_A53x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2_A53x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2_A53x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A73x2_A53x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A73x2_A53x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x2_A53x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x2_A53x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x2_A53x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2_A53x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A73x2_A53x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A73x2_A53x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2_A53x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x2_A53x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A73x2_A53x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A73x2_A53x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x2_A53x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x2_A53x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A73x2_A53x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x2_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x2_A53x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2_A53x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A73x2_A53x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x2_A53x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x2_A53x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A73x2_A53x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A73x2_A53x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x2_A53x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x2_A53x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A73x2_A53x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x2_A53x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A73x2_A53x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2_A53x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2_A53x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2_A53x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x2_A53x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A73x2_A53x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A73x2_A53x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x2_A53x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A73x2_A53x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A73x2_A53x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A73x2_A53x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A73x2_A53x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2_A53x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A73x2_A53x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2_A53x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A73x2_A53x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2_A53x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A73x2_A53x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A73x2_A53x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x2_A53x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x2_A53x4.cluster0	Cluster_ARM_Cortex-A73	ARM Cortex-A73 Cluster CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu0	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu1	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2_A53x4.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.

Table 3-48 FVP_Base_Cortex-A73x2-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu2	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu3	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x2_A53x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A73x2_A53x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A73x2_A53x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A73x2_A53x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A73x2_A53x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.49 FVP_Base_Cortex-A73x4

FVP_Base_Cortex-A73x4 contains the following instances:

Table 3-49 FVP_Base_Cortex-A73x4 instances

Name	Type	Description
FVP_Base_Cortex_A73x4	FVP_Base_Cortex_A73x4	Base Platform Compute Subsystem for ARMCortexA73x4CT.
FVP_Base_Cortex_A73x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A73x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A73x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A73x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A73x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A73x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A73x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A73x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A73x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A73x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A73x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A73x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A73x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A73x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A73x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A73x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A73x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A73x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A73x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A73x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A73x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A73x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A73x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A73x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A73x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A73x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A73x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A73x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A73x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4.cluster0	Cluster_ARM_Cortex-A73	ARM Cortex-A73 Cluster CT model.
FVP_Base_Cortex_A73x4.cluster0.cpu0	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0.cpu1	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0.cpu2	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0.cpu3	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.

Table 3-49 FVP_Base_Cortex-A73x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A73x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A73x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A73x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.50 FVP_Base_Cortex-A73x4-A53x4

FVP_Base_Cortex-A73x4-A53x4 contains the following instances:

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4	FVP_Base_Cortex_A73x4_A53x4	Base Platform Compute Subsystem for ARMCortexA73x4CT and ARMCortexA53x4CT.
FVP_Base_Cortex_A73x4_A53x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4_A53x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A73x4_A53x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A73x4_A53x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x4_A53x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A73x4_A53x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A73x4_A53x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4_A53x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4_A53x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4_A53x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4_A53x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A73x4_A53x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A73x4_A53x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x4_A53x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x4_A53x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x4_A53x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4_A53x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A73x4_A53x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A73x4_A53x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4_A53x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A73x4_A53x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A73x4_A53x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A73x4_A53x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x4_A53x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A73x4_A53x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A73x4_A53x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A73x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A73x4_A53x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A73x4_A53x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4_A53x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A73x4_A53x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x4_A53x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A73x4_A53x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A73x4_A53x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A73x4_A53x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A73x4_A53x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A73x4_A53x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A73x4_A53x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x4_A53x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A73x4_A53x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4_A53x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4_A53x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4_A53x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A73x4_A53x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A73x4_A53x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A73x4_A53x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A73x4_A53x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A73x4_A53x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A73x4_A53x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A73x4_A53x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A73x4_A53x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4_A53x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A73x4_A53x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4_A53x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A73x4_A53x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4_A53x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A73x4_A53x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A73x4_A53x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A73x4_A53x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A73x4_A53x4.cluster0	Cluster_ARM_Cortex-A73	ARM Cortex-A73 Cluster CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu0	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu1	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu2	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu3	ARM_Cortex-A73	ARM Cortex-A73 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4_A53x4.cluster1	Cluster_ARM_Cortex-A53	ARM Cortex-A53 Cluster CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu0	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu1	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu2	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu3	ARM_Cortex-A53	ARM Cortex-A53 CT model.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1.cpu3.l1icache	*PVCache*	PV Cache.

Table 3-50 FVP_Base_Cortex-A73x4-A53x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A73x4_A53x4.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A73x4_A53x4.cluster1_labeller	*Labeller*	-
FVP_Base_Cortex_A73x4_A53x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A73x4_A53x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A73x4_A53x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A73x4_A53x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.51 FVP_Base_Cortex-A75

FVP_Base_Cortex-A75 contains the following instances:

Table 3-51 FVP_Base_Cortex-A75 instances

Name	Type	Description
FVP_Base_Cortex_A75	FVP_Base_Cortex_A75	Base Platform Compute Subsystem for ARMCortexA75CT.
FVP_Base_Cortex_A75.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A75.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A75.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A75.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A75.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A75.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A75.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A75.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A75.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A75.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A75.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A75.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A75.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A75.bp.pl111_clcd.pl111x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A75.bp.pl111_clcd.pl111x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75.bp.pl111_clcd.pl111x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75.bp.pl111_clcd.pl111x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75.bp.pl111_clcd.pl111x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A75.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A75.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A75.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A75.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A75.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A75.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A75.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A75.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A75.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A75.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A75.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A75.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A75.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A75.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A75.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A75.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A75.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A75.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A75.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A75.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A75.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A75.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A75.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.cluster0	Cluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A75.cluster0.cpu0	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu1	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu2	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu3	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A75.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A75.elfloader	*ElfLoader*	ELF loader component.

Table 3-51 FVP_Base_Cortex-A75 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A75.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.52 FVP_Base_Cortex-A75x1

FVP_Base_Cortex-A75x1 contains the following instances:

Table 3-52 FVP_Base_Cortex-A75x1 instances

Name	Type	Description
FVP_Base_Cortex_A75x1	FVP_Base_Cortex_A75x1	Base Platform Compute Subsystem for ARMCortexA75x1CT.
FVP_Base_Cortex_A75x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A75x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A75x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-52 FVP_Base_Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-52 FVP_Base_Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A75x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A75x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A75x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-52 FVP_Base_Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A75x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A75x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A75x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A75x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A75x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-52 FVP_Base_Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A75x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A75x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A75x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A75x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-52 FVP_Base_Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A75x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A75x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A75x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A75x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-52 FVP_Base_Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A75x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A75x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A75x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A75x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A75x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A75x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A75x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A75x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A75x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A75x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A75x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A75x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A75x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A75x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-52 FVP_Base_Cortex-A75x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A75x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x1.cluster0	Cluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A75x1.cluster0.cpu0	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A75x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A75x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A75x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A75x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.53 FVP_Base_Cortex-A75x2

FVP_Base_Cortex-A75x2 contains the following instances:

Table 3-53 FVP_Base_Cortex-A75x2 instances

Name	Type	Description
FVP_Base_Cortex_A75x2	FVP_Base_Cortex_A75x2	Base Platform Compute Subsystem for ARMCortexA75x2CT.
FVP_Base_Cortex_A75x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A75x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A75x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-53 FVP_Base_Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-53 FVP_Base_Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A75x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A75x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A75x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-53 FVP_Base_Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A75x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A75x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A75x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A75x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A75x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-53 FVP_Base_Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A75x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A75x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A75x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A75x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-53 FVP_Base_Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A75x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A75x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A75x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A75x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-53 FVP_Base_Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A75x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A75x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A75x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A75x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A75x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A75x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A75x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A75x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A75x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A75x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A75x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A75x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A75x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A75x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-53 FVP_Base_Cortex-A75x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A75x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x2.cluster0	Cluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A75x2.cluster0.cpu0	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x2.cluster0.cpu1	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A75x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A75x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A75x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A75x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.54 FVP_Base_Cortex-A75x4

FVP_Base_Cortex-A75x4 contains the following instances:

Table 3-54 FVP_Base_Cortex-A75x4 instances

Name	Type	Description
FVP_Base_Cortex_A75x4	FVP_Base_Cortex_A75x4	Base Platform Compute Subsystem for ARMCortexA75x4CT.
FVP_Base_Cortex_A75x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A75x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A75x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A75x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A75x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A75x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A75x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A75x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A75x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A75x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A75x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A75x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A75x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A75x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A75x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A75x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A75x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A75x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A75x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A75x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A75x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A75x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A75x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A75x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A75x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A75x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A75x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A75x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A75x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A75x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A75x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A75x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A75x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A75x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A75x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A75x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A75x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A75x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A75x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A75x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A75x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A75x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A75x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A75x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A75x4.cluster0	Cluster_ARM_Cortex-A75	ARM Cortex-A75 Cluster CT model.
FVP_Base_Cortex_A75x4.cluster0.cpu0	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu1	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu2	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.

Table 3-54 FVP_Base_Cortex-A75x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A75x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu3	ARM_Cortex-A75	ARM Cortex-A75 CT model.
FVP_Base_Cortex_A75x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A75x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A75x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A75x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A75x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A75x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A75x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.55 FVP_Base_Cortex-A76AEx2

FVP_Base_Cortex-A76AEx2 contains the following instances:

Table 3-55 FVP_Base_Cortex-A76AEx2 instances

Name	Type	Description
FVP_Base_Cortex_A76AEx2	FVP_Base_Cortex_A76AEx2	Base Platform Compute Subsystem for ARMCortexA76AEx2CT.
FVP_Base_Cortex_A76AEx2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A76AEx2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76AEx2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76AEx2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A76AEx2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76AEx2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76AEx2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A76AEx2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76AEx2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76AEx2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A76AEx2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A76AEx2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76AEx2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76AEx2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A76AEx2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A76AEx2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A76AEx2.bp.nontrusteddrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76AEx2.bp.nontrusteddromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76AEx2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76AEx2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76AEx2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A76AEx2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A76AEx2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A76AEx2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A76AEx2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76AEx2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A76AEx2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A76AEx2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A76AEx2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A76AEx2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A76AEx2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A76AEx2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.securefla sh	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.bp.securefla shloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx2.bp.smsc_91c 111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A76AEx2.bp.sp805_w dog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76AEx2.bp.sp810_sy sctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A76AEx2.bp.sp810_sy sctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.sp810_sy sctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.sp810_sy sctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.sp810_sy sctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.bp.terminal_ 0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx2.bp.terminal_ 1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx2.bp.terminal_ 2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx2.bp.terminal_ 3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx2.bp.trusted_k ey_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A76AEx2.bp.trusted_n v_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A76AEx2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76AEx2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A76AEx2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A76AEx2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A76AEx2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A76AEx2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A76AEx2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A76AEx2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A76AEx2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A76AEx2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A76AEx2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-55 FVP_Base_Cortex-A76AEx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx2.cluster0	Cluster_ARM_Cortex-A76AE	ARM Cortex-A76AE Cluster CT model.
FVP_Base_Cortex_A76AEx2.cluster0.cpu0	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A76AEx2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76AEx2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx2.cluster0.cpu1	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A76AEx2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76AEx2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A76AEx2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A76AEx2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A76AEx2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.56 FVP_Base_Cortex-A76AEx4

FVP_Base_Cortex-A76AEx4 contains the following instances:

Table 3-56 FVP_Base_Cortex-A76AEx4 instances

Name	Type	Description
FVP_Base_Cortex_A76AEx4	FVP_Base_Cortex_A76AEx4	Base Platform Compute Subsystem for ARMCortexA76AEx4CT.
FVP_Base_Cortex_A76AEx4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A76AEx4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76AEx4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76AEx4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76AEx4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_Base_Cortex_A76AEx4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76AEx4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76AEx4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A76AEx4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76AEx4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76AEx4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A76AEx4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A76AEx4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76AEx4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76AEx4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A76AEx4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A76AEx4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A76AEx4.bp.nontrusteddrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76AEx4.bp.nontrusteddromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76AEx4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76AEx4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76AEx4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A76AEx4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A76AEx4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A76AEx4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A76AEx4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76AEx4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A76AEx4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A76AEx4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A76AEx4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A76AEx4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A76AEx4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A76AEx4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.securefla sh	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.bp.securefla shloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76AEx4.bp.smsc_91c 111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A76AEx4.bp.sp805_w dog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76AEx4.bp.sp810_sy sctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A76AEx4.bp.sp810_sy sctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.sp810_sy sctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.sp810_sy sctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.sp810_sy sctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.bp.terminal_ 0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx4.bp.terminal_ 1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx4.bp.terminal_ 2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx4.bp.terminal_ 3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76AEx4.bp.trusted_k ey_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A76AEx4.bp.trusted_n v_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A76AEx4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76AEx4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A76AEx4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A76AEx4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A76AEx4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A76AEx4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A76AEx4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A76AEx4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A76AEx4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A76AEx4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76AEx4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76AEx4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A76AEx4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.cluster0	Cluster_ARM_Cortex-A76AE	ARM Cortex-A76AE Cluster CT model.
FVP_Base_Cortex_A76AEx4.cluster0.cpu0	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A76AEx4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76AEx4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu1	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A76AEx4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76AEx4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu2	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A76AEx4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76AEx4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu3	ARM_Cortex-A76AE	ARM Cortex-A76AE CT model.
FVP_Base_Cortex_A76AEx4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76AEx4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76AEx4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.

Table 3-56 FVP_Base_Cortex-A76AEx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76AEx4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A76AEx4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A76AEx4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A76AEx4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A76AEx4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.57 FVP_Base_Cortex-A76x1

FVP_Base_Cortex-A76x1 contains the following instances:

Table 3-57 FVP_Base_Cortex-A76x1 instances

Name	Type	Description
FVP_Base_Cortex_A76x1	FVP_Base_Cortex_A76x1	Base Platform Compute Subsystem for ARMCortexA76x1CT.
FVP_Base_Cortex_A76x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A76x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A76x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-57 FVP_Base_Cortex-A76x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-57 FVP_Base_Cortex-A76x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A76x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A76x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A76x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-57 FVP_Base_Cortex-A76x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A76x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A76x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A76x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A76x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A76x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-57 FVP_Base_Cortex-A76x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A76x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A76x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A76x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A76x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A76x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A76x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-57 FVP_Base_Cortex-A76x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A76x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A76x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A76x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A76x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A76x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-57 FVP_Base_Cortex-A76x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A76x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A76x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A76x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A76x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A76x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A76x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A76x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A76x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A76x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A76x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A76x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A76x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A76x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A76x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-57 FVP_Base_Cortex-A76x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A76x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x1.cluster0	Cluster_ARM_Cortex-A76	ARM Cortex-A76 Cluster CT model.
FVP_Base_Cortex_A76x1.cluster0.cpu0	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A76x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A76x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A76x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A76x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A76x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.58 FVP_Base_Cortex-A76x2

FVP_Base_Cortex-A76x2 contains the following instances:

Table 3-58 FVP_Base_Cortex-A76x2 instances

Name	Type	Description
FVP_Base_Cortex_A76x2	FVP_Base_Cortex_A76x2	Base Platform Compute Subsystem for ARMCortexA76x2CT.
FVP_Base_Cortex_A76x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A76x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A76x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-58 FVP_Base_Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-58 FVP_Base_Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A76x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A76x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A76x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-58 FVP_Base_Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A76x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A76x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A76x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A76x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A76x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-58 FVP_Base_Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A76x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A76x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A76x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A76x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-58 FVP_Base_Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A76x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A76x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A76x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A76x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-58 FVP_Base_Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A76x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A76x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A76x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A76x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A76x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A76x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A76x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A76x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A76x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A76x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A76x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A76x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A76x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A76x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-58 FVP_Base_Cortex-A76x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A76x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x2.cluster0	Cluster_ARM_Cortex-A76	ARM Cortex-A76 Cluster CT model.
FVP_Base_Cortex_A76x2.cluster0.cpu0	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A76x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x2.cluster0.cpu1	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A76x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A76x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A76x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A76x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A76x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.59 FVP_Base_Cortex-A76x4

FVP_Base_Cortex-A76x4 contains the following instances:

Table 3-59 FVP_Base_Cortex-A76x4 instances

Name	Type	Description
FVP_Base_Cortex_A76x4	FVP_Base_Cortex_A76x4	Base Platform Compute Subsystem for ARMCortexA76x4CT.
FVP_Base_Cortex_A76x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A76x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A76x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A76x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A76x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A76x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A76x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A76x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A76x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A76x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A76x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A76x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A76x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A76x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A76x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A76x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A76x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A76x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A76x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A76x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A76x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A76x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A76x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A76x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A76x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A76x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A76x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A76x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A76x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A76x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A76x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A76x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A76x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A76x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A76x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A76x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A76x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A76x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A76x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A76x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A76x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A76x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A76x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A76x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A76x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A76x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A76x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A76x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A76x4.cluster0	Cluster_ARM_Cortex-A76	ARM Cortex-A76 Cluster CT model.
FVP_Base_Cortex_A76x4.cluster0.cpu0	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A76x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu1	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A76x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu2	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A76x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.

Table 3-59 FVP_Base_Cortex-A76x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A76x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu3	ARM_Cortex-A76	ARM Cortex-A76 CT model.
FVP_Base_Cortex_A76x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A76x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A76x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A76x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A76x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A76x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A76x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.60 FVP_Base_Cortex-A77x1

FVP_Base_Cortex-A77x1 contains the following instances:

Table 3-60 FVP_Base_Cortex-A77x1 instances

Name	Type	Description
FVP_Base_Cortex_A77x1	FVP_Base_Cortex_A77x1	Base Platform Compute Subsystem for ARMCortexA77x1CT.
FVP_Base_Cortex_A77x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A77x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A77x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A77x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A77x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-60 FVP_Base_Cortex-A77x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-60 FVP_Base_Cortex-A77x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A77x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A77x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A77x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A77x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A77x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A77x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-60 FVP_Base_Cortex-A77x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A77x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A77x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A77x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A77x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A77x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-60 FVP_Base_Cortex-A77x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A77x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A77x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A77x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A77x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A77x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A77x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A77x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-60 FVP_Base_Cortex-A77x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A77x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A77x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A77x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A77x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A77x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A77x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-60 FVP_Base_Cortex-A77x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A77x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A77x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A77x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A77x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A77x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A77x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A77x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A77x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A77x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A77x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A77x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A77x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A77x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A77x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A77x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-60 FVP_Base_Cortex-A77x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A77x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x1.cluster0	Cluster_ARM_Cortex-A77	ARM Cortex-A77 Cluster CT model.
FVP_Base_Cortex_A77x1.cluster0.cpu0	ARM_Cortex-A77	ARM Cortex-A77 CT model.
FVP_Base_Cortex_A77x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A77x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A77x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A77x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A77x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A77x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.61 FVP_Base_Cortex-A77x2

FVP_Base_Cortex-A77x2 contains the following instances:

Table 3-61 FVP_Base_Cortex-A77x2 instances

Name	Type	Description
FVP_Base_Cortex_A77x2	FVP_Base_Cortex_A77x2	Base Platform Compute Subsystem for ARMCortexA77x2CT.
FVP_Base_Cortex_A77x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A77x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A77x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A77x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A77x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-61 FVP_Base_Cortex-A77x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-61 FVP_Base_Cortex-A77x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A77x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A77x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A77x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A77x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A77x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A77x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-61 FVP_Base_Cortex-A77x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A77x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A77x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A77x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A77x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A77x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-61 FVP_Base_Cortex-A77x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A77x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A77x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A77x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A77x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A77x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A77x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A77x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-61 FVP_Base_Cortex-A77x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A77x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A77x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A77x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A77x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A77x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A77x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-61 FVP_Base_Cortex-A77x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A77x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A77x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A77x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A77x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A77x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A77x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A77x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A77x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A77x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A77x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A77x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A77x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A77x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A77x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A77x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-61 FVP_Base_Cortex-A77x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A77x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x2.cluster0	Cluster_ARM_Cortex-A77	ARM Cortex-A77 Cluster CT model.
FVP_Base_Cortex_A77x2.cluster0.cpu0	ARM_Cortex-A77	ARM Cortex-A77 CT model.
FVP_Base_Cortex_A77x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A77x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x2.cluster0.cpu1	ARM_Cortex-A77	ARM Cortex-A77 CT model.
FVP_Base_Cortex_A77x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A77x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A77x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A77x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A77x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A77x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.62 FVP_Base_Cortex-A77x4

FVP_Base_Cortex-A77x4 contains the following instances:

Table 3-62 FVP_Base_Cortex-A77x4 instances

Name	Type	Description
FVP_Base_Cortex_A77x4	FVP_Base_Cortex_A77x4	Base Platform Compute Subsystem for ARMCortexA77x4CT.
FVP_Base_Cortex_A77x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A77x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A77x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A77x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A77x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A77x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A77x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A77x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A77x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A77x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A77x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A77x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A77x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A77x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A77x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A77x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A77x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A77x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A77x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A77x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A77x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A77x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A77x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A77x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A77x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A77x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A77x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A77x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A77x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A77x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A77x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A77x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A77x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A77x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A77x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A77x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A77x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A77x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A77x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A77x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A77x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A77x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A77x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A77x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A77x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A77x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A77x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A77x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A77x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A77x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A77x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A77x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A77x4.cluster0	Cluster_ARM_Cortex-A77	ARM Cortex-A77 Cluster CT model.
FVP_Base_Cortex_A77x4.cluster0.cpu0	ARM_Cortex-A77	ARM Cortex-A77 CT model.
FVP_Base_Cortex_A77x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A77x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu1	ARM_Cortex-A77	ARM Cortex-A77 CT model.
FVP_Base_Cortex_A77x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A77x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu2	ARM_Cortex-A77	ARM Cortex-A77 CT model.
FVP_Base_Cortex_A77x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A77x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.

Table 3-62 FVP_Base_Cortex-A77x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A77x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu3	ARM_Cortex-A77	ARM Cortex-A77 CT model.
FVP_Base_Cortex_A77x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A77x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A77x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A77x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A77x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A77x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A77x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.63 FVP_Base_Cortex-A78Cx1

FVP_Base_Cortex-A78Cx1 contains the following instances:

Table 3-63 FVP_Base_Cortex-A78Cx1 instances

Name	Type	Description
FVP_Base_Cortex_A78Cx1	FVP_Base_Cortex_A78Cx1	Base Platform Compute Subsystem for ARMCortexA78Cx1CT.
FVP_Base_Cortex_A78Cx1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A78Cx1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78Cx1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78Cx1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A78Cx1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-63 FVP_Base_Cortex-A78Cx1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-63 FVP_Base_Cortex-A78Cx1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78Cx1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78Cx1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A78Cx1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A78Cx1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A78Cx1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-63 FVP_Base_Cortex-A78Cx1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78Cx1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78Cx1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A78Cx1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A78Cx1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-63 FVP_Base_Cortex-A78Cx1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A78Cx1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A78Cx1.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78Cx1.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78Cx1.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A78Cx1.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A78Cx1.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx1.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx1.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78Cx1.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78Cx1.bp.pl111_clcd _labeller	*Labeller*	-

Table 3-63 FVP_Base_Cortex-A78Cx1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A78Cx1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78Cx1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78Cx1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A78Cx1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A78Cx1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A78Cx1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78Cx1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A78Cx1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-63 FVP_Base_Cortex-A78Cx1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A78Cx1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A78Cx1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A78Cx1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78Cx1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A78Cx1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A78Cx1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A78Cx1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A78Cx1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A78Cx1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A78Cx1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A78Cx1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 3-63 FVP_Base_Cortex-A78Cx1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A78Cx1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx1.cluster0	Cluster_ARM_Cortex-A78C	ARM Cortex-A78C Cluster CT model.
FVP_Base_Cortex_A78Cx1.cluster0.cpu0	ARM_Cortex-A78C	ARM Cortex-A78C CT model.
FVP_Base_Cortex_A78Cx1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78Cx1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A78Cx1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A78Cx1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A78Cx1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.64 FVP_Base_Cortex-A78Cx2

FVP_Base_Cortex-A78Cx2 contains the following instances:

Table 3-64 FVP_Base_Cortex-A78Cx2 instances

Name	Type	Description
FVP_Base_Cortex_A78Cx2	FVP_Base_Cortex_A78Cx2	Base Platform Compute Subsystem for ARMCortexA78Cx2CT.
FVP_Base_Cortex_A78Cx2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A78Cx2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78Cx2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78Cx2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A78Cx2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78Cx2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78Cx2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A78Cx2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A78Cx2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A78Cx2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78Cx2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78Cx2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A78Cx2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A78Cx2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A78Cx2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A78Cx2.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78Cx2.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78Cx2.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A78Cx2.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A78Cx2.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx2.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx2.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78Cx2.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78Cx2.bp.pl111_clcd _labeller	*Labeller*	-

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A78Cx2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78Cx2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78Cx2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A78Cx2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A78Cx2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A78Cx2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78Cx2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A78Cx2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A78Cx2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A78Cx2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A78Cx2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78Cx2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A78Cx2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A78Cx2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A78Cx2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A78Cx2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A78Cx2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A78Cx2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A78Cx2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A78Cx2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx2.cluster0	Cluster_ARM_Cortex-A78C	ARM Cortex-A78C Cluster CT model.
FVP_Base_Cortex_A78Cx2.cluster0.cpu0	ARM_Cortex-A78C	ARM Cortex-A78C CT model.
FVP_Base_Cortex_A78Cx2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78Cx2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx2.cluster0.cpu1	ARM_Cortex-A78C	ARM Cortex-A78C CT model.
FVP_Base_Cortex_A78Cx2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78Cx2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx2.cluster0_labeller	*Labeller*	-

Table 3-64 FVP_Base_Cortex-A78Cx2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A78Cx2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A78Cx2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A78Cx2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.65 FVP_Base_Cortex-A78Cx4

FVP_Base_Cortex-A78Cx4 contains the following instances:

Table 3-65 FVP_Base_Cortex-A78Cx4 instances

Name	Type	Description
FVP_Base_Cortex_A78Cx4	FVP_Base_Cortex_A78Cx4	Base Platform Compute Subsystem for ARMCortexA78Cx4CT.
FVP_Base_Cortex_A78Cx4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A78Cx4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78Cx4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78Cx4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78Cx4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A78Cx4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78Cx4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78Cx4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A78Cx4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A78Cx4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A78Cx4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78Cx4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78Cx4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A78Cx4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A78Cx4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78Cx4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A78Cx4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A78Cx4.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78Cx4.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78Cx4.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A78Cx4.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A78Cx4.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx4.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78Cx4.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78Cx4.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78Cx4.bp.pl111_clcd _labeller	*Labeller*	-

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A78Cx4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78Cx4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78Cx4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A78Cx4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A78Cx4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78Cx4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78Cx4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A78Cx4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78Cx4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A78Cx4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78Cx4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A78Cx4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A78Cx4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A78Cx4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78Cx4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A78Cx4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A78Cx4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A78Cx4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A78Cx4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A78Cx4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A78Cx4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A78Cx4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78Cx4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A78Cx4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78Cx4.cluster0	Cluster_ARM_Cortex-A78C	ARM Cortex-A78C Cluster CT model.
FVP_Base_Cortex_A78Cx4.cluster0.cpu0	ARM_Cortex-A78C	ARM Cortex-A78C CT model.
FVP_Base_Cortex_A78Cx4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78Cx4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu1	ARM_Cortex-A78C	ARM Cortex-A78C CT model.
FVP_Base_Cortex_A78Cx4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78Cx4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu2	ARM_Cortex-A78C	ARM Cortex-A78C CT model.
FVP_Base_Cortex_A78Cx4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-65 FVP_Base_Cortex-A78Cx4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78Cx4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu3	ARM_Cortex-A78C	ARM Cortex-A78C CT model.
FVP_Base_Cortex_A78Cx4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78Cx4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78Cx4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A78Cx4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A78Cx4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A78Cx4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A78Cx4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.66 FVP_Base_Cortex-A78x1

FVP_Base_Cortex-A78x1 contains the following instances:

Table 3-66 FVP_Base_Cortex-A78x1 instances

Name	Type	Description
FVP_Base_Cortex_A78x1	FVP_Base_Cortex_A78x1	Base Platform Compute Subsystem for ARMCortexA78x1CT.
FVP_Base_Cortex_A78x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A78x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A78x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-66 FVP_Base_Cortex-A78x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-66 FVP_Base_Cortex-A78x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A78x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A78x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A78x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-66 FVP_Base_Cortex-A78x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A78x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A78x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A78x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A78x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A78x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-66 FVP_Base_Cortex-A78x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A78x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A78x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A78x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A78x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-66 FVP_Base_Cortex-A78x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A78x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A78x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A78x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A78x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-66 FVP_Base_Cortex-A78x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A78x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A78x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A78x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A78x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A78x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A78x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A78x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A78x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A78x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A78x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A78x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A78x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A78x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A78x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-66 FVP_Base_Cortex-A78x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A78x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x1.cluster0	Cluster_ARM_Cortex-A78	ARM Cortex-A78 Cluster CT model.
FVP_Base_Cortex_A78x1.cluster0.cpu0	ARM_Cortex-A78	ARM Cortex-A78 CT model.
FVP_Base_Cortex_A78x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A78x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A78x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A78x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A78x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.67 FVP_Base_Cortex-A78x2

FVP_Base_Cortex-A78x2 contains the following instances:

Table 3-67 FVP_Base_Cortex-A78x2 instances

Name	Type	Description
FVP_Base_Cortex_A78x2	FVP_Base_Cortex_A78x2	Base Platform Compute Subsystem for ARMCortexA78x2CT.
FVP_Base_Cortex_A78x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A78x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A78x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-67 FVP_Base_Cortex-A78x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-67 FVP_Base_Cortex-A78x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A78x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A78x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A78x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-67 FVP_Base_Cortex-A78x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A78x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A78x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A78x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A78x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A78x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-67 FVP_Base_Cortex-A78x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A78x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A78x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A78x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A78x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-67 FVP_Base_Cortex-A78x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A78x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A78x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A78x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A78x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-67 FVP_Base_Cortex-A78x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A78x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A78x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A78x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A78x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A78x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A78x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A78x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A78x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A78x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A78x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A78x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A78x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A78x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A78x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-67 FVP_Base_Cortex-A78x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A78x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x2.cluster0	Cluster_ARM_Cortex-A78	ARM Cortex-A78 Cluster CT model.
FVP_Base_Cortex_A78x2.cluster0.cpu0	ARM_Cortex-A78	ARM Cortex-A78 CT model.
FVP_Base_Cortex_A78x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x2.cluster0.cpu1	ARM_Cortex-A78	ARM Cortex-A78 CT model.
FVP_Base_Cortex_A78x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A78x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A78x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A78x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A78x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.68 FVP_Base_Cortex-A78x4

FVP_Base_Cortex-A78x4 contains the following instances:

Table 3-68 FVP_Base_Cortex-A78x4 instances

Name	Type	Description
FVP_Base_Cortex_A78x4	FVP_Base_Cortex_A78x4	Base Platform Compute Subsystem for ARMCortexA78x4CT.
FVP_Base_Cortex_A78x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_A78x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_A78x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_A78x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_A78x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_A78x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_A78x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_A78x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_A78x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_A78x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_A78x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_A78x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_A78x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_A78x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_A78x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_A78x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_A78x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_A78x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_A78x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_A78x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_A78x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_A78x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_A78x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_A78x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_A78x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_A78x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_A78x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_A78x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_A78x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_A78x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_A78x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_A78x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_A78x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_A78x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_A78x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_A78x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_A78x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_A78x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_A78x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_A78x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_A78x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_A78x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_A78x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_A78x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_A78x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_A78x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_A78x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Cortex_A78x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_A78x4.cluster0	Cluster_ARM_Cortex-A78	ARM Cortex-A78 Cluster CT model.
FVP_Base_Cortex_A78x4.cluster0.cpu0	ARM_Cortex-A78	ARM Cortex-A78 CT model.
FVP_Base_Cortex_A78x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu1	ARM_Cortex-A78	ARM Cortex-A78 CT model.
FVP_Base_Cortex_A78x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu2	ARM_Cortex-A78	ARM Cortex-A78 CT model.
FVP_Base_Cortex_A78x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.

Table 3-68 FVP_Base_Cortex-A78x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_A78x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu3	ARM_Cortex-A78	ARM Cortex-A78 CT model.
FVP_Base_Cortex_A78x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_A78x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_A78x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_A78x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_A78x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_A78x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_A78x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.69 FVP_Base_Cortex-X1x1

FVP_Base_Cortex-X1x1 contains the following instances:

Table 3-69 FVP_Base_Cortex-X1x1 instances

Name	Type	Description
FVP_Base_Cortex_X1x1	FVP_Base_Cortex_X1x1	Base Platform Compute Subsystem for ARMCortexX1x1CT.
FVP_Base_Cortex_X1x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_X1x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_X1x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_X1x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_X1x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-69 FVP_Base_Cortex-X1x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-69 FVP_Base_Cortex-X1x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_X1x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_X1x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_X1x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_X1x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_X1x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_X1x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-69 FVP_Base_Cortex-X1x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_X1x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_X1x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_X1x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_X1x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_X1x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-69 FVP_Base_Cortex-X1x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_X1x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_X1x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_X1x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_X1x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_X1x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_X1x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_X1x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-69 FVP_Base_Cortex-X1x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_X1x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_X1x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_X1x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_X1x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_X1x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_X1x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.

Table 3-69 FVP_Base_Cortex-X1x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_X1x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_X1x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_X1x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_X1x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_X1x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_X1x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_X1x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_X1x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_X1x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_X1x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_X1x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_X1x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_X1x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_X1x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_X1x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-69 FVP_Base_Cortex-X1x1 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x1.cluster0	Cluster_ARM_Cortex-X1	ARM Cortex-X1 Cluster CT model.
FVP_Base_Cortex_X1x1.cluster0.cpu0	ARM_Cortex-X1	ARM Cortex-X1 CT model.
FVP_Base_Cortex_X1x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_X1x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x1.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_X1x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_X1x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_X1x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_X1x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.70 FVP_Base_Cortex-X1x2

FVP_Base_Cortex-X1x2 contains the following instances:

Table 3-70 FVP_Base_Cortex-X1x2 instances

Name	Type	Description
FVP_Base_Cortex_X1x2	FVP_Base_Cortex_X1x2	Base Platform Compute Subsystem for ARMCortexX1x2CT.
FVP_Base_Cortex_X1x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_X1x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_X1x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_X1x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_X1x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-70 FVP_Base_Cortex-X1x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-70 FVP_Base_Cortex-X1x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_X1x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_X1x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_X1x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_X1x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_X1x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_X1x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-70 FVP_Base_Cortex-X1x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_X1x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_X1x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_X1x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_X1x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_X1x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-70 FVP_Base_Cortex-X1x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_X1x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_X1x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_X1x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_X1x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_X1x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_X1x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_X1x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-70 FVP_Base_Cortex-X1x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_X1x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_X1x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_X1x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_X1x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_X1x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_X1x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.

Table 3-70 FVP_Base_Cortex-X1x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_X1x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_X1x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_X1x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_X1x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_X1x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_X1x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_X1x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_X1x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_X1x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_X1x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_X1x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_X1x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_X1x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_X1x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_X1x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-70 FVP_Base_Cortex-X1x2 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x2.cluster0	Cluster_ARM_Cortex-X1	ARM Cortex-X1 Cluster CT model.
FVP_Base_Cortex_X1x2.cluster0.cpu0	ARM_Cortex-X1	ARM Cortex-X1 CT model.
FVP_Base_Cortex_X1x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_X1x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x2.cluster0.cpu1	ARM_Cortex-X1	ARM Cortex-X1 CT model.
FVP_Base_Cortex_X1x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_X1x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x2.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_X1x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_X1x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_X1x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_X1x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.71 FVP_Base_Cortex-X1x4

FVP_Base_Cortex-X1x4 contains the following instances:

Table 3-71 FVP_Base_Cortex-X1x4 instances

Name	Type	Description
FVP_Base_Cortex_X1x4	FVP_Base_Cortex_X1x4	Base Platform Compute Subsystem for ARMCortexX1x4CT.
FVP_Base_Cortex_X1x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Cortex_X1x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_X1x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Cortex_X1x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Cortex_X1x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Cortex_X1x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_X1x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Cortex_X1x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Cortex_X1x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Cortex_X1x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Cortex_X1x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_X1x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Cortex_X1x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Cortex_X1x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Cortex_X1x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Cortex_X1x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Cortex_X1x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Cortex_X1x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Cortex_X1x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Cortex_X1x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Cortex_X1x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Cortex_X1x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Cortex_X1x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Cortex_X1x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Cortex_X1x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Cortex_X1x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Cortex_X1x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Cortex_X1x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Cortex_X1x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Cortex_X1x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Cortex_X1x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Cortex_X1x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_X1x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Cortex_X1x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Cortex_X1x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Cortex_X1x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Cortex_X1x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Cortex_X1x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Cortex_X1x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Cortex_X1x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Cortex_X1x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Cortex_X1x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Cortex_X1x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Cortex_X1x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Cortex_X1x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Cortex_X1x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Cortex_X1x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Cortex_X1x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Cortex_X1x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Cortex_X1x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Cortex_X1x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Cortex_X1x4.cluster0	Cluster_ARM_Cortex-X1	ARM Cortex-X1 Cluster CT model.
FVP_Base_Cortex_X1x4.cluster0.cpu0	ARM_Cortex-X1	ARM Cortex-X1 CT model.
FVP_Base_Cortex_X1x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_X1x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu1	ARM_Cortex-X1	ARM Cortex-X1 CT model.
FVP_Base_Cortex_X1x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_X1x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu2	ARM_Cortex-X1	ARM Cortex-X1 CT model.
FVP_Base_Cortex_X1x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_X1x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu3	ARM_Cortex-X1	ARM Cortex-X1 CT model.
FVP_Base_Cortex_X1x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Cortex_X1x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.

Table 3-71 FVP_Base_Cortex-X1x4 instances (continued)

Name	Type	Description
FVP_Base_Cortex_X1x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Cortex_X1x4.cluster0_labeller	*Labeller*	-
FVP_Base_Cortex_X1x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Cortex_X1x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Cortex_X1x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Cortex_X1x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.72 FVP_Base_Neoverse-E1x1

FVP_Base_Neoverse-E1x1 contains the following instances:

Table 3-72 FVP_Base_Neoverse-E1x1 instances

Name	Type	Description
FVP_Base_Neoverse_E1x1	FVP_Base_Neoverse_E1x1	Base Platform Compute Subsystem for ARMNeoverseE1x1CT.
FVP_Base_Neoverse_E1x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Neoverse_E1x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_E1x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_E1x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Neoverse_E1x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-72 FVP_Base_Neoverse-E1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-72 FVP_Base_Neoverse-E1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_E1x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_E1x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Neoverse_E1x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Neoverse_E1x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Neoverse_E1x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-72 FVP_Base_Neoverse-E1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_E1x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_E1x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Neoverse_E1x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Neoverse_E1x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Neoverse_E1x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Neoverse_E1x1.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-72 FVP_Base_Neoverse-E1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x1.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_E1x1.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Neoverse_E1x1.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Neoverse_E1x1.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x1.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_E1x1.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_E1x1.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Neoverse_E1x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-72 FVP_Base_Neoverse-E1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_E1x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Neoverse_E1x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Neoverse_E1x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Neoverse_E1x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_E1x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Neoverse_E1x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-72 FVP_Base_Neoverse-E1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Neoverse_E1x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Neoverse_E1x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Neoverse_E1x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_E1x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Neoverse_E1x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Neoverse_E1x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Neoverse_E1x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Neoverse_E1x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Neoverse_E1x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Neoverse_E1x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Neoverse_E1x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Neoverse_E1x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-72 FVP_Base_Neoverse-E1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Neoverse_E1x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x1.cluster0	Cluster_ARM_Neoverse-E1	ARM Neoverse-E1 Cluster CT model.
FVP_Base_Neoverse_E1x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_E1x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x1.cluster0.cpu0.thread0	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x1.cluster0.cpu0.thread1	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x1.cluster0_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Neoverse_E1x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Neoverse_E1x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Neoverse_E1x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.73 FVP_Base_Neoverse-E1x2

FVP_Base_Neoverse-E1x2 contains the following instances:

Table 3-73 FVP_Base_Neoverse-E1x2 instances

Name	Type	Description
FVP_Base_Neoverse_E1x2	FVP_Base_Neoverse_E1x2	Base Platform Compute Subsystem for ARMNeoverseE1x2CT.
FVP_Base_Neoverse_E1x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Neoverse_E1x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_E1x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_E1x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Neoverse_E1x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_E1x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_E1x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Neoverse_E1x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Neoverse_E1x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Neoverse_E1x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_E1x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_E1x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Neoverse_E1x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Neoverse_E1x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Neoverse_E1x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Neoverse_E1x2.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_E1x2.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Neoverse_E1x2.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Neoverse_E1x2.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x2.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_E1x2.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_E1x2.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Neoverse_E1x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_E1x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Neoverse_E1x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Neoverse_E1x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Neoverse_E1x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_E1x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Neoverse_E1x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Neoverse_E1x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Neoverse_E1x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Neoverse_E1x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_E1x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Neoverse_E1x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Neoverse_E1x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Neoverse_E1x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Neoverse_E1x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Neoverse_E1x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Neoverse_E1x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Neoverse_E1x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Neoverse_E1x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Neoverse_E1x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x2.cluster0	Cluster_ARM_Neoverse-E1	ARM Neoverse-E1 Cluster CT model.
FVP_Base_Neoverse_E1x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_E1x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x2.cluster0.cpu0.thread0	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x2.cluster0.cpu0.thread1	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_E1x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x2.cluster0.cpu1.thread0	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x2.cluster0.cpu1.thread1	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x2.cluster0_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x2.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-73 FVP_Base_Neoverse-E1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Neoverse_E1x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Neoverse_E1x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.74 FVP_Base_Neoverse-E1x4

FVP_Base_Neoverse-E1x4 contains the following instances:

Table 3-74 FVP_Base_Neoverse-E1x4 instances

Name	Type	Description
FVP_Base_Neoverse_E1x4	FVP_Base_Neoverse_E1x4	Base Platform Compute Subsystem for ARMNeoverseE1x4CT.
FVP_Base_Neoverse_E1x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Neoverse_E1x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_E1x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_E1x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_E1x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Neoverse_E1x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_E1x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_E1x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Neoverse_E1x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Neoverse_E1x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Neoverse_E1x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_E1x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_E1x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Neoverse_E1x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Neoverse_E1x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_E1x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Neoverse_E1x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Neoverse_E1x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_E1x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Neoverse_E1x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Neoverse_E1x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_E1x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_E1x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_E1x4.bp.pl111_clcd_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Neoverse_E1x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_E1x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Neoverse_E1x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Neoverse_E1x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_E1x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_E1x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Neoverse_E1x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_E1x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Neoverse_E1x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_E1x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Neoverse_E1x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Neoverse_E1x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Neoverse_E1x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_E1x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Neoverse_E1x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Neoverse_E1x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Neoverse_E1x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Neoverse_E1x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Neoverse_E1x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Neoverse_E1x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Neoverse_E1x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_Base_Neoverse_E1x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_E1x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Neoverse_E1x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_E1x4.cluster0	Cluster_ARM_Neoverse-E1	ARM Neoverse-E1 Cluster CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_E1x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu0.thread0	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu0.thread1	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_E1x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu1.thread0	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu1.thread1	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-74 FVP_Base_Neoverse-E1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_E1x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu2.thread0	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu2.thread1	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_E1x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_E1x4.cluster0.cpu3.thread0	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0.cpu3.thread1	ARM_Neoverse-E1	ARM Neoverse-E1 CT model.
FVP_Base_Neoverse_E1x4.cluster0_labeller	*Labeller*	-
FVP_Base_Neoverse_E1x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Neoverse_E1x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Neoverse_E1x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Neoverse_E1x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.75 FVP_Base_Neoverse-N1x1

FVP_Base_Neoverse-N1x1 contains the following instances:

Table 3-75 FVP_Base_Neoverse-N1x1 instances

Name	Type	Description
FVP_Base_Neoverse_N1x1	FVP_Base_Neoverse_N1x1	Base Platform Compute Subsystem for ARMNeoverseN1x1CT.
FVP_Base_Neoverse_N1x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Neoverse_N1x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_N1x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_N1x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Neoverse_N1x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-75 FVP_Base_Neoverse-N1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-75 FVP_Base_Neoverse-N1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_N1x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_N1x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Neoverse_N1x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Neoverse_N1x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Neoverse_N1x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-75 FVP_Base_Neoverse-N1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_N1x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_N1x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Neoverse_N1x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Neoverse_N1x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-75 FVP_Base_Neoverse-N1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Neoverse_N1x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Neoverse_N1x1.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_N1x1.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_N1x1.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Neoverse_N1x1.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Neoverse_N1x1.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x1.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x1.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_N1x1.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_N1x1.bp.pl111_clcd _labeller	*Labeller*	-

Table 3-75 FVP_Base_Neoverse-N1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Neoverse_N1x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_N1x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_N1x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Neoverse_N1x1.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Neoverse_N1x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Neoverse_N1x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_N1x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Neoverse_N1x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-75 FVP_Base_Neoverse-N1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Neoverse_N1x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Neoverse_N1x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Neoverse_N1x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_N1x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Neoverse_N1x1.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Neoverse_N1x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Neoverse_N1x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Neoverse_N1x1.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Neoverse_N1x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Neoverse_N1x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Neoverse_N1x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 3-75 FVP_Base_Neoverse-N1x1 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Neoverse_N1x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x1.cluster0	Cluster_ARM_Neoverse-N1	ARM Neoverse-N1 Cluster CT model.
FVP_Base_Neoverse_N1x1.cluster0.cpu0	ARM_Neoverse-N1	ARM Neoverse-N1 CT model.
FVP_Base_Neoverse_N1x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_N1x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x1.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x1.cluster0_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Neoverse_N1x1.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Neoverse_N1x1.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Neoverse_N1x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.76 FVP_Base_Neoverse-N1x2

FVP_Base_Neoverse-N1x2 contains the following instances:

Table 3-76 FVP_Base_Neoverse-N1x2 instances

Name	Type	Description
FVP_Base_Neoverse_N1x2	FVP_Base_Neoverse_N1x2	Base Platform Compute Subsystem for ARMNeoverseN1x2CT.
FVP_Base_Neoverse_N1x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Neoverse_N1x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_N1x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_N1x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Neoverse_N1x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_N1x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_N1x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Neoverse_N1x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Neoverse_N1x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Neoverse_N1x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_N1x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_N1x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Neoverse_N1x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Neoverse_N1x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Neoverse_N1x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Neoverse_N1x2.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_N1x2.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_N1x2.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Neoverse_N1x2.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Neoverse_N1x2.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x2.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x2.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_N1x2.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_N1x2.bp.pl111_clcd _labeller	*Labeller*	-

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Neoverse_N1x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_N1x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_N1x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Neoverse_N1x2.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Neoverse_N1x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Neoverse_N1x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_N1x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Neoverse_N1x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Neoverse_N1x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Neoverse_N1x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Neoverse_N1x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_N1x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Neoverse_N1x2.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Neoverse_N1x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Neoverse_N1x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Neoverse_N1x2.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Neoverse_N1x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Neoverse_N1x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Neoverse_N1x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Neoverse_N1x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x2.cluster0	Cluster_ARM_Neoverse-N1	ARM Neoverse-N1 Cluster CT model.
FVP_Base_Neoverse_N1x2.cluster0.cpu0	ARM_Neoverse-N1	ARM Neoverse-N1 CT model.
FVP_Base_Neoverse_N1x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_N1x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x2.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x2.cluster0.cpu1	ARM_Neoverse-N1	ARM Neoverse-N1 CT model.
FVP_Base_Neoverse_N1x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_N1x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x2.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x2.cluster0_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x2.dapmemlogger	*PVBusLogger*	Bus Logger.

Table 3-76 FVP_Base_Neoverse-N1x2 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x2.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Neoverse_N1x2.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Neoverse_N1x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

3.77 FVP_Base_Neoverse-N1x4

FVP_Base_Neoverse-N1x4 contains the following instances:

Table 3-77 FVP_Base_Neoverse-N1x4 instances

Name	Type	Description
FVP_Base_Neoverse_N1x4	FVP_Base_Neoverse_N1x4	Base Platform Compute Subsystem for ARMNeoverseN1x4CT.
FVP_Base_Neoverse_N1x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_Base_Neoverse_N1x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_N1x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_Base_Neoverse_N1x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_Base_Neoverse_N1x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_Base_Neoverse_N1x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_N1x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_Base_Neoverse_N1x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_Base_Neoverse_N1x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_Base_Neoverse_N1x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_Base_Neoverse_N1x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_N1x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_N1x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_Base_Neoverse_N1x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_Base_Neoverse_N1x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_Base_Neoverse_N1x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_Base_Neoverse_N1x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_Base_Neoverse_N1x4.bp.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_N1x4.bp.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_Base_Neoverse_N1x4.bp.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_Base_Neoverse_N1x4.bp.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_Base_Neoverse_N1x4.bp.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x4.bp.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_Base_Neoverse_N1x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_Base_Neoverse_N1x4.bp.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_Base_Neoverse_N1x4.bp.pl111_clcd_labeller	*Labeller*	-

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_Base_Neoverse_N1x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_N1x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_Base_Neoverse_N1x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_Base_Neoverse_N1x4.bp.reset_or	*OrGate*	Or Gate.
FVP_Base_Neoverse_N1x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_Base_Neoverse_N1x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_Base_Neoverse_N1x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_Base_Neoverse_N1x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_N1x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_Base_Neoverse_N1x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_Base_Neoverse_N1x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_Base_Neoverse_N1x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_Base_Neoverse_N1x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_Base_Neoverse_N1x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_Base_Neoverse_N1x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_Base_Neoverse_N1x4.bp.ve_sysregs	VE_SysRegs	-
FVP_Base_Neoverse_N1x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_Base_Neoverse_N1x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_Base_Neoverse_N1x4.bp.virtio_net_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_Base_Neoverse_N1x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_Base_Neoverse_N1x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_Base_Neoverse_N1x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_Base_Neoverse_N1x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_Base_Neoverse_N1x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_Base_Neoverse_N1x4.cluster0	Cluster_ARM_Neoverse-N1	ARM Neoverse-N1 Cluster CT model.
FVP_Base_Neoverse_N1x4.cluster0.cpu0	ARM_Neoverse-N1	ARM Neoverse-N1 CT model.
FVP_Base_Neoverse_N1x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_N1x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu0.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu1	ARM_Neoverse-N1	ARM Neoverse-N1 CT model.
FVP_Base_Neoverse_N1x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_N1x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu1.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu2	ARM_Neoverse-N1	ARM Neoverse-N1 CT model.
FVP_Base_Neoverse_N1x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.

Table 3-77 FVP_Base_Neoverse-N1x4 instances (continued)

Name	Type	Description
FVP_Base_Neoverse_N1x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu2.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu3	ARM_Neoverse-N1	ARM Neoverse-N1 CT model.
FVP_Base_Neoverse_N1x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_Base_Neoverse_N1x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0.cpu3.l2cache	*PVCache*	PV Cache.
FVP_Base_Neoverse_N1x4.cluster0_labeller	*Labeller*	-
FVP_Base_Neoverse_N1x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_Base_Neoverse_N1x4.elfloader	*ElfLoader*	ELF loader component.
FVP_Base_Neoverse_N1x4.gic_distributor	*GIC_IRI*	GIC Interrupt Redistribution Infrastucture component.
FVP_Base_Neoverse_N1x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

Chapter 4
BaseR Platform FVPs

This chapter lists the BaseR Platform FVPs and the instances in them.

For the BaseR Platform memory map, see *BaseR Platform memory map* in the Fast Models Reference Manual.

It contains the following sections:

- *4.1 FVP_BaseR_Cortex-R52x1* on page 4-729.
- *4.2 FVP_BaseR_Cortex-R52x2* on page 4-737.
- *4.3 FVP_BaseR_Cortex-R52x4* on page 4-746.

4.1 FVP_BaseR_Cortex-R52x1

FVP_BaseR_Cortex-R52x1 contains the following instances:

Table 4-1 FVP_BaseR_Cortex-R52x1 instances

Name	Type	Description
FVP_BaseR_Cortex_R52x1	FVP_BaseR_Cortex_R52x1	Base Platform Compute Subsystem for ARMCortexR52x1CT.
FVP_BaseR_Cortex_R52x1.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_BaseR_Cortex_R52x1.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_BaseR_Cortex_R52x1.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x1.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x1.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_BaseR_Cortex_R52x1.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x1.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x1.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_BaseR_Cortex_R52x1.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 4-1 FVP_BaseR_Cortex-R52x1 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x1.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-1 FVP_BaseR_Cortex-R52x1 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x1.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_BaseR_Cortex_R52x1.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_BaseR_Cortex_R52x1.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_BaseR_Cortex_R52x1.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x1.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x1.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x1.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x1.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_BaseR_Cortex_R52x1.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_BaseR_Cortex_R52x1.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x1.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 4-1 FVP_BaseR_Cortex-R52x1 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x1.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_BaseR_Cortex_R52x1.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_BaseR_Cortex_R52x1.bp.hdlcd0_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x1.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_BaseR_Cortex_R52x1.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_BaseR_Cortex_R52x1.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x1.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x1.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x1.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x1.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x1.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x1.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-1 FVP_BaseR_Cortex-R52x1 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x1.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_BaseR_Cortex_R52x1.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_BaseR_Cortex_R52x1.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_BaseR_Cortex_R52x1.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_BaseR_Cortex_R52x1.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_BaseR_Cortex_R52x1.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_BaseR_Cortex_R52x1.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x1.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x1.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_BaseR_Cortex_R52x1.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_BaseR_Cortex_R52x1.bp.pl111_clcd _labeller	*Labeller*	-

Table 4-1 FVP_BaseR_Cortex-R52x1 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x1.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_BaseR_Cortex_R52x1.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_BaseR_Cortex_R52x1.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_BaseR_Cortex_R52x1.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_BaseR_Cortex_R52x1.bp.reset_or	*OrGate*	Or Gate.
FVP_BaseR_Cortex_R52x1.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x1.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x1.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_BaseR_Cortex_R52x1.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_BaseR_Cortex_R52x1.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_BaseR_Cortex_R52x1.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-1 FVP_BaseR_Cortex-R52x1 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x1.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x1.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x1.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x1.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x1.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_BaseR_Cortex_R52x1.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_BaseR_Cortex_R52x1.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_BaseR_Cortex_R52x1.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_BaseR_Cortex_R52x1.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_BaseR_Cortex_R52x1.bp.ve_sysregs	VE_SysRegs	-
FVP_BaseR_Cortex_R52x1.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_BaseR_Cortex_R52x1.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_BaseR_Cortex_R52x1.bp.virtio_net_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x1.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_BaseR_Cortex_R52x1.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x1.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_BaseR_Cortex_R52x1.bp.virtiop9device_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x1.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_BaseR_Cortex_R52x1.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 4-1 FVP_BaseR_Cortex-R52x1 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x1.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_BaseR_Cortex_R52x1.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x1.cluster0	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x1.cluster0.cpu0	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x1.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_BaseR_Cortex_R52x1.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x1.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x1.cluster0.gic_iri	gic_iri	GIC IRI internal to cluster.
FVP_BaseR_Cortex_R52x1.cluster0_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x1.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_BaseR_Cortex_R52x1.elfloader	*ElfLoader*	ELF loader component.
FVP_BaseR_Cortex_R52x1.flash_ram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x1.pctl	*Base_PowerController*	Base Platforms Power Controller.

4.2 FVP_BaseR_Cortex-R52x2

FVP_BaseR_Cortex-R52x2 contains the following instances:

Table 4-2 FVP_BaseR_Cortex-R52x2 instances

Name	Type	Description
FVP_BaseR_Cortex_R52x2	FVP_BaseR_Cortex_R52x2	Base Platform Compute Subsystem for ARMCortexR52x2CT.
FVP_BaseR_Cortex_R52x2.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_BaseR_Cortex_R52x2.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_BaseR_Cortex_R52x2.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x2.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x2.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_BaseR_Cortex_R52x2.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x2.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x2.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_BaseR_Cortex_R52x2.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_BaseR_Cortex_R52x2.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_BaseR_Cortex_R52x2.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_BaseR_Cortex_R52x2.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x2.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x2.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x2.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x2.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_BaseR_Cortex_R52x2.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_BaseR_Cortex_R52x2.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x2.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_BaseR_Cortex_R52x2.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_BaseR_Cortex_R52x2.bp.hdlcd0_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x2.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_BaseR_Cortex_R52x2.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_BaseR_Cortex_R52x2.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x2.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x2.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x2.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x2.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x2.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x2.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_BaseR_Cortex_R52x2.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_BaseR_Cortex_R52x2.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_BaseR_Cortex_R52x2.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_BaseR_Cortex_R52x2.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_BaseR_Cortex_R52x2.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_BaseR_Cortex_R52x2.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x2.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x2.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_BaseR_Cortex_R52x2.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_BaseR_Cortex_R52x2.bp.pl111_clcd _labeller	*Labeller*	-

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_BaseR_Cortex_R52x2.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_BaseR_Cortex_R52x2.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_BaseR_Cortex_R52x2.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_BaseR_Cortex_R52x2.bp.reset_or	*OrGate*	Or Gate.
FVP_BaseR_Cortex_R52x2.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x2.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x2.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_BaseR_Cortex_R52x2.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_BaseR_Cortex_R52x2.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_BaseR_Cortex_R52x2.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x2.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x2.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x2.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x2.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_BaseR_Cortex_R52x2.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_BaseR_Cortex_R52x2.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_BaseR_Cortex_R52x2.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_BaseR_Cortex_R52x2.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_BaseR_Cortex_R52x2.bp.ve_sysregs	VE_SysRegs	-
FVP_BaseR_Cortex_R52x2.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_BaseR_Cortex_R52x2.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_BaseR_Cortex_R52x2.bp.virtio_net_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x2.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_BaseR_Cortex_R52x2.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x2.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_BaseR_Cortex_R52x2.bp.virtiop9device_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x2.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_BaseR_Cortex_R52x2.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_BaseR_Cortex_R52x2.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x2.cluster0	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x2.cluster0.cpu0	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x2.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_BaseR_Cortex_R52x2.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x2.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x2.cluster0.cpu1	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x2.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_BaseR_Cortex_R52x2.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x2.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x2.cluster0.gic_iri	gic_iri	GIC IRI internal to cluster.
FVP_BaseR_Cortex_R52x2.cluster0_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x2.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_BaseR_Cortex_R52x2.elfloader	*ElfLoader*	ELF loader component.

Table 4-2 FVP_BaseR_Cortex-R52x2 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x2.flash_ram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x2.pctl	*Base_PowerController*	Base Platforms Power Controller.

4.3 FVP_BaseR_Cortex-R52x4

FVP_BaseR_Cortex-R52x4 contains the following instances:

Table 4-3 FVP_BaseR_Cortex-R52x4 instances

Name	Type	Description
FVP_BaseR_Cortex_R52x4	FVP_BaseR_Cortex_R52x4	Base Platform Compute Subsystem for ARMCortexR52x4CT.
FVP_BaseR_Cortex_R52x4.bp	BasePlatformPeripherals	Peripherals and address map for the Base Platform.
FVP_BaseR_Cortex_R52x4.bp.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_BaseR_Cortex_R52x4.bp.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x4.bp.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x4.bp.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_BaseR_Cortex_R52x4.bp.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x4.bp.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_BaseR_Cortex_R52x4.bp.ap_refclk	*MemoryMappedGenericTimer*	ARM Generic Timer.
FVP_BaseR_Cortex_R52x4.bp.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.bp.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.clock300MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.clock32KHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.bp.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.dram_alias_warning	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_BaseR_Cortex_R52x4.bp.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_BaseR_Cortex_R52x4.bp.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_BaseR_Cortex_R52x4.bp.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x4.bp.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x4.bp.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x4.bp.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x4.bp.generic_watchdog	*MemoryMappedGenericWatchdog*	ARM Generic Watchdog.
FVP_BaseR_Cortex_R52x4.bp.hdlcd0	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_BaseR_Cortex_R52x4.bp.hdlcd0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x4.bp.hdlcd0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.bp.hdlcd0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_BaseR_Cortex_R52x4.bp.hdlcd0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_BaseR_Cortex_R52x4.bp.hdlcd0_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x4.bp.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_BaseR_Cortex_R52x4.bp.mmc	*MMC*	Generic Multimedia Card.
FVP_BaseR_Cortex_R52x4.bp.nontrustedrom	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x4.bp.nontrustedromloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x4.bp.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x4.bp.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x4.bp.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x4.bp.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_BaseR_Cortex_R52x4.bp.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.bp.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_BaseR_Cortex_R52x4.bp.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_BaseR_Cortex_R52x4.bp.pl050_kmi 0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_BaseR_Cortex_R52x4.bp.pl050_kmi 0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.pl050_kmi 1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_BaseR_Cortex_R52x4.bp.pl050_kmi 1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_BaseR_Cortex_R52x4.bp.pl111_clcd. pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_BaseR_Cortex_R52x4.bp.pl111_clcd. pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x4.bp.pl111_clcd. pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_BaseR_Cortex_R52x4.bp.pl111_clcd. pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_BaseR_Cortex_R52x4.bp.pl111_clcd. pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_BaseR_Cortex_R52x4.bp.pl111_clcd _labeller	*Labeller*	-

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.bp.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_BaseR_Cortex_R52x4.bp.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_BaseR_Cortex_R52x4.bp.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_BaseR_Cortex_R52x4.bp.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.refcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_BaseR_Cortex_R52x4.bp.reset_or	*OrGate*	Or Gate.
FVP_BaseR_Cortex_R52x4.bp.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.secureflash	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_BaseR_Cortex_R52x4.bp.secureflashloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_BaseR_Cortex_R52x4.bp.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_BaseR_Cortex_R52x4.bp.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_BaseR_Cortex_R52x4.bp.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_BaseR_Cortex_R52x4.bp.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.bp.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.bp.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x4.bp.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x4.bp.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x4.bp.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_BaseR_Cortex_R52x4.bp.trusted_key_storage	*RootKeyStorage*	Trusted Root-Key Storage unit.
FVP_BaseR_Cortex_R52x4.bp.trusted_nv_counter	*NonVolatileCounter*	Trusted Non-Volatile Counter unit.
FVP_BaseR_Cortex_R52x4.bp.trusted_rng	*RandomNumberGenerator*	Random Number Generator unit.
FVP_BaseR_Cortex_R52x4.bp.trusted_watchdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_BaseR_Cortex_R52x4.bp.tzc_400	*TZC_400*	TrustZone Address Space Controller.
FVP_BaseR_Cortex_R52x4.bp.ve_sysregs	VE_SysRegs	-
FVP_BaseR_Cortex_R52x4.bp.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_BaseR_Cortex_R52x4.bp.virtio_net	*VirtioNetMMIO*	VirtioNet device over MMIO transport.
FVP_BaseR_Cortex_R52x4.bp.virtio_net_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x4.bp.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_BaseR_Cortex_R52x4.bp.virtioblockdevice_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x4.bp.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_BaseR_Cortex_R52x4.bp.virtiop9device_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x4.bp.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_BaseR_Cortex_R52x4.bp.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.bp.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.bp.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_BaseR_Cortex_R52x4.clockdivider0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_BaseR_Cortex_R52x4.cluster0	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x4.cluster0.cpu0	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x4.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_BaseR_Cortex_R52x4.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.cpu1	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x4.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_BaseR_Cortex_R52x4.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.cpu1.l1icache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.cpu2	ARM_CortexR52	ARM CortexR52 MP CT model.
FVP_BaseR_Cortex_R52x4.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_BaseR_Cortex_R52x4.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.cpu3	ARM_CortexR52	ARM CortexR52 MP CT model.

Table 4-3 FVP_BaseR_Cortex-R52x4 instances (continued)

Name	Type	Description
FVP_BaseR_Cortex_R52x4.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_BaseR_Cortex_R52x4.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_BaseR_Cortex_R52x4.cluster0.gic_iri	gic_iri	GIC IRI internal to cluster.
FVP_BaseR_Cortex_R52x4.cluster0_labeller	*Labeller*	-
FVP_BaseR_Cortex_R52x4.dapmemlogger	*PVBusLogger*	Bus Logger.
FVP_BaseR_Cortex_R52x4.elfloader	*ElfLoader*	ELF loader component.
FVP_BaseR_Cortex_R52x4.flash_ram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_BaseR_Cortex_R52x4.pctl	*Base_PowerController*	Base Platforms Power Controller.

Chapter 5
VE Platform FVPs

This chapter lists the VE Platform FVPs and the instances in them.

For the VE memory maps, see *VE memory map for Cortex-A series* and *VE memory map for Cortex-R series* in the Fast Models Reference Manual.

It contains the following sections:

- *5.1 FVP_VE_Cortex-A15x1* on page 5-757.
- *5.2 FVP_VE_Cortex-A15x1-A7x1* on page 5-765.
- *5.3 FVP_VE_Cortex-A15x2* on page 5-774.
- *5.4 FVP_VE_Cortex-A15x2-A7x2* on page 5-782.
- *5.5 FVP_VE_Cortex-A15x4* on page 5-792.
- *5.6 FVP_VE_Cortex-A15x4-A7x4* on page 5-801.
- *5.7 FVP_VE_Cortex-A17x1* on page 5-812.
- *5.8 FVP_VE_Cortex-A17x1-A7x1* on page 5-820.
- *5.9 FVP_VE_Cortex-A17x2* on page 5-829.
- *5.10 FVP_VE_Cortex-A17x4* on page 5-837.
- *5.11 FVP_VE_Cortex-A17x4-A7x4* on page 5-846.
- *5.12 FVP_VE_Cortex-A5x1* on page 5-857.
- *5.13 FVP_VE_Cortex-A5x2* on page 5-865.
- *5.14 FVP_VE_Cortex-A5x4* on page 5-873.
- *5.15 FVP_VE_Cortex-A7x1* on page 5-882.
- *5.16 FVP_VE_Cortex-A7x2* on page 5-890.
- *5.17 FVP_VE_Cortex-A7x4* on page 5-898.
- *5.18 FVP_VE_Cortex-A9x1* on page 5-907.
- *5.19 FVP_VE_Cortex-A9x2* on page 5-915.
- *5.20 FVP_VE_Cortex-A9x4* on page 5-923.
- *5.21 FVP_VE_Cortex-R4* on page 5-932.

- *5.22 FVP_VE_Cortex-R5x1* on page 5-940.
- *5.23 FVP_VE_Cortex-R5x2* on page 5-948.
- *5.24 FVP_VE_Cortex-R7x1* on page 5-956.
- *5.25 FVP_VE_Cortex-R7x2* on page 5-963.
- *5.26 FVP_VE_Cortex-R8x1* on page 5-970.
- *5.27 FVP_VE_Cortex-R8x2* on page 5-977.
- *5.28 FVP_VE_Cortex-R8x4* on page 5-984.

5.1 FVP_VE_Cortex-A15x1

FVP_VE_Cortex-A15x1 contains the following instances:

Table 5-1 FVP_VE_Cortex-A15x1 instances

Name	Type	Description
FVP_VE_Cortex_A15x1	FVP_VE_Cortex_A15x1	Top level component of the Cortex_A15x1 Versatile Express inspired model.
FVP_VE_Cortex_A15x1.cluster	Cluster_ARM_Cortex-A15	ARM Cortex-A15 Cluster CT model.
FVP_VE_Cortex_A15x1.cluster.cpu0	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x1.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x1.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x1.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A15x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.

Table 5-1 FVP_VE_Cortex-A15x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A15x1.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A15x1.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x1.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x1.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A15x1.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x1.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x1.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).

Table 5-1 FVP_VE_Cortex-A15x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A15x1.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A15x1.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x1.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x1.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 5-1 FVP_VE_Cortex-A15x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A15x1.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x1.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x1.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x1.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-1 FVP_VE_Cortex-A15x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A15x1.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A15x1.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A15x1.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A15x1.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-1 FVP_VE_Cortex-A15x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x1.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A15x1.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A15x1.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x1.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A15x1.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 5-1 FVP_VE_Cortex-A15x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x1.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A15x1.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A15x1.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A15x1.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x1.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x1.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x1.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 5-1 FVP_VE_Cortex-A15x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A15x1.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A15x1.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A15x1.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A15x1.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A15x1.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.2 FVP_VE_Cortex-A15x1-A7x1

FVP_VE_Cortex-A15x1-A7x1 contains the following instances:

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1	FVP_VE_Cortex_A15x1_A7x1	Top level component of the Cortex A15x1 A7x1 Versatile Express inspired model.
FVP_VE_Cortex_A15x1_A7x1.coretile	ARM_Cortex_A15x1_A7x1_CT	Dual cluster ARM Cortex-A15x1 and ARM Cortex-A7x1 Core Tile.
FVP_VE_Cortex_A15x1_A7x1.coretile.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_VE_Cortex_A15x1_A7x1.coretile.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.coretile.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster0	Cluster_ARM_Cortex-A15	ARM Cortex-A15 Cluster CT model.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster0.cpu0	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster0.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster1	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster1.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster1.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1_A7x1.coretile.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x1_A7x1.coretile.dualclustersystemconfigurationblock	*DualClusterSystemConfigurationBlock*	Dual Cluster System Configuration Block.
FVP_VE_Cortex_A15x1_A7x1.coretile.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A15x1_A7x1.coretile.v7_vgic	*v7_VGIC*	System VGIC architecture version v7.
FVP_VE_Cortex_A15x1_A7x1.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1_A7x1.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1_A7x1.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A15x1_A7x1.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A15x1_A7x1.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1_A7x1.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x1_A7x1.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A15x1_A7x1.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A15x1_A7x1.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x1_A7x1.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x1_A7x1.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x1_A7x1.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x1_A7x1.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x1_A7x1.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A15x1_A7x1.motherboard.mmc	*MMC*	Generic Multimedia Card.

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x1_A7x1.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A15x1_A7x1.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x1_A7x1.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x1_A7x1.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A15x1_A7x1.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A15x1_A7x1.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A15x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x1_A7x1.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x1_A7x1.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x1_A7x1.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x1_A7x1.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A15x1_A7x1.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A15x1_A7x1.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.

Table 5-2 FVP_VE_Cortex-A15x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x1_A7x1.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A15x1_A7x1.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A15x1_A7x1.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x1_A7x1.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.3 FVP_VE_Cortex-A15x2

FVP_VE_Cortex-A15x2 contains the following instances:

Table 5-3 FVP_VE_Cortex-A15x2 instances

Name	Type	Description
FVP_VE_Cortex_A15x2	FVP_VE_Cortex_A15x2	Top level component of the Cortex_A15x2 Versatile Express inspired model.
FVP_VE_Cortex_A15x2.cluster	Cluster_ARM_Cortex-A15	ARM Cortex-A15 Cluster CT model.
FVP_VE_Cortex_A15x2.cluster.cpu0	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x2.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2.cluster.cpu1	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x2.cluster.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2.cluster.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A15x2.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-3 FVP_VE_Cortex-A15x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A15x2.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A15x2.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x2.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x2.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A15x2.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x2.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x2.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-3 FVP_VE_Cortex-A15x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A15x2.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A15x2.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A15x2.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x2.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x2.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x2.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x2.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x2.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 5-3 FVP_VE_Cortex-A15x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A15x2.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A15x2.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x2.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x2.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-3 FVP_VE_Cortex-A15x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x2.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A15x2.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A15x2.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x2.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x2.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x2.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x2.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A15x2.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A15x2.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-3 FVP_VE_Cortex-A15x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x2.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A15x2.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A15x2.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x2.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A15x2.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 5-3 FVP_VE_Cortex-A15x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x2.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A15x2.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A15x2.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A15x2.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x2.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x2.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x2.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 5-3 FVP_VE_Cortex-A15x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A15x2.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A15x2.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A15x2.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A15x2.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A15x2.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.4 FVP_VE_Cortex-A15x2-A7x2

FVP_VE_Cortex-A15x2-A7x2 contains the following instances:

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2	FVP_VE_Cortex_A15x2_A7x2	Top level component of the Cortex A15x2 A7x2 Versatile Express inspired model.
FVP_VE_Cortex_A15x2_A7x2.coretile	ARM_Cortex_A15x2_A7x2_CT	Dual cluster ARM Cortex-A15x2 and ARM Cortex-A7x2 Core Tile.
FVP_VE_Cortex_A15x2_A7x2.coretile.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_VE_Cortex_A15x2_A7x2.coretile.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.coretile.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0	Cluster_ARM_Cortex-A15	ARM Cortex-A15 Cluster CT model.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu0	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu1	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.cpu1.l1icache	*PVCache*	PV Cache.

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu1	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x2_A7x2.coretile.dualclustersystemconfigurationblock	*DualClusterSystemConfigurationBlock*	Dual Cluster System Configuration Block.
FVP_VE_Cortex_A15x2_A7x2.coretile.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A15x2_A7x2.coretile.v7_vgic	*v7_VGIC*	System VGIC architecture version v7.
FVP_VE_Cortex_A15x2_A7x2.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A15x2_A7x2.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2_A7x2.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2_A7x2.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A15x2_A7x2.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A15x2_A7x2.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x2_A7x2.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x2_A7x2.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A15x2_A7x2.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A15x2_A7x2.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x2_A7x2.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x2_A7x2.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x2_A7x2.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x2_A7x2.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A15x2_A7x2.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x2_A7x2.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A15x2_A7x2.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x2_A7x2.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x2_A7x2.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x2_A7x2.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A15x2_A7x2.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A15x2_A7x2.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x2_A7x2.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x2_A7x2.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x2_A7x2.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x2_A7x2.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A15x2_A7x2.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A15x2_A7x2.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A15x2_A7x2.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A15x2_A7x2.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-4 FVP_VE_Cortex-A15x2-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x2_A7x2.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x2_A7x2.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.5 FVP_VE_Cortex-A15x4

FVP_VE_Cortex-A15x4 contains the following instances:

Table 5-5 FVP_VE_Cortex-A15x4 instances

Name	Type	Description
FVP_VE_Cortex_A15x4	FVP_VE_Cortex_A15x4	Top level component of the Cortex_A15x4 Versatile Express inspired model.
FVP_VE_Cortex_A15x4.cluster	Cluster_ARM_Cortex-A15	ARM Cortex-A15 Cluster CT model.
FVP_VE_Cortex_A15x4.cluster.cpu0	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.cpu1	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4.cluster.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.cpu2	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4.cluster.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu2.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.cpu3	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4.cluster.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu3.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4.cluster.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A15x4.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A15x4.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x4.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x4.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x4.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A15x4.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x4.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x4.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A15x4.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A15x4.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A15x4.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x4.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x4.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A15x4.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A15x4.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x4.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x4.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x4.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x4.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A15x4.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A15x4.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x4.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x4.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x4.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A15x4.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A15x4.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x4.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A15x4.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A15x4.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x4.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A15x4.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x4.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x4.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A15x4.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A15x4.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A15x4.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A15x4.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A15x4.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A15x4.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.

Table 5-5 FVP_VE_Cortex-A15x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A15x4.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.6 FVP_VE_Cortex-A15x4-A7x4

FVP_VE_Cortex-A15x4-A7x4 contains the following instances:

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4	FVP_VE_Cortex_A15x4_A7x4	Top level component of the Cortex A15x4 A7x4 Versatile Express inspired model.
FVP_VE_Cortex_A15x4_A7x4.coretile	ARM_Cortex_A15x4_A7x4_CT	Dual cluster ARM Cortex-A15x4 and ARM Cortex-A7x4 Core Tile.
FVP_VE_Cortex_A15x4_A7x4.coretile.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_VE_Cortex_A15x4_A7x4.coretile.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.coretile.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0	Cluster_ARM_Cortex-A15	ARM Cortex-A15 Cluster CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu0	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu1	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu1.l1icache	*PVCache*	PV Cache.

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu2	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu2.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu3	ARM_Cortex-A15	ARM Cortex-A15 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu3.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu1	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu2	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu2.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu3	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu3.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A15x4_A7x4.coretile.dualclustersystemconfigurationblock	*DualClusterSystemConfigurationBlock*	Dual Cluster System Configuration Block.
FVP_VE_Cortex_A15x4_A7x4.coretile.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A15x4_A7x4.coretile.v7_vgic	*v7_VGIC*	System VGIC architecture version v7.
FVP_VE_Cortex_A15x4_A7x4.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A15x4_A7x4.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x4_A7x4.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x4_A7x4.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A15x4_A7x4.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A15x4_A7x4.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4_A7x4.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A15x4_A7x4.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A15x4_A7x4.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A15x4_A7x4.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x4_A7x4.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A15x4_A7x4.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x4_A7x4.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A15x4_A7x4.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A15x4_A7x4.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A15x4_A7x4.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A15x4_A7x4.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x4_A7x4.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A15x4_A7x4.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A15x4_A7x4.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A15x4_A7x4.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A15x4_A7x4.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4_A7x4.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4_A7x4.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4_A7x4.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A15x4_A7x4.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A15x4_A7x4.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A15x4_A7x4.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A15x4_A7x4.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A15x4_A7x4.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-6 FVP_VE_Cortex-A15x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A15x4_A7x4.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A15x4_A7x4.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.7 FVP_VE_Cortex-A17x1

FVP_VE_Cortex-A17x1 contains the following instances:

Table 5-7 FVP_VE_Cortex-A17x1 instances

Name	Type	Description
FVP_VE_Cortex_A17x1	FVP_VE_Cortex_A17x1	Top level component of the Cortex_A17x1 Versatile Express inspired model.
FVP_VE_Cortex_A17x1.cluster	Cluster_ARM_Cortex-A17	ARM Cortex-A17 Cluster CT model.
FVP_VE_Cortex_A17x1.cluster.cpu0	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x1.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x1.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x1.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A17x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.

Table 5-7 FVP_VE_Cortex-A17x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A17x1.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A17x1.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x1.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x1.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A17x1.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x1.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x1.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).

Table 5-7 FVP_VE_Cortex-A17x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1.gic400	*GIC_400*	GIC-400 Generic Interrupt Controller.
FVP_VE_Cortex_A17x1.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A17x1.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A17x1.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x1.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x1.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 5-7 FVP_VE_Cortex-A17x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A17x1.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x1.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x1.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x1.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-7 FVP_VE_Cortex-A17x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A17x1.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A17x1.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A17x1.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A17x1.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-7 FVP_VE_Cortex-A17x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A17x1.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A17x1.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A17x1.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x1.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A17x1.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 5-7 FVP_VE_Cortex-A17x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x1.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A17x1.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A17x1.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A17x1.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x1.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x1.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x1.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 5-7 FVP_VE_Cortex-A17x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A17x1.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A17x1.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A17x1.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A17x1.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A17x1.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.8 FVP_VE_Cortex-A17x1-A7x1

FVP_VE_Cortex-A17x1-A7x1 contains the following instances:

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1	FVP_VE_Cortex_A17x1_A7x1	Top level component of the Cortex A17x1 A7x1 Versatile Express inspired model.
FVP_VE_Cortex_A17x1_A7x1.coretile	ARM_Cortex_A17x1_A7x1_CT	Dual cluster ARM Cortex-A17x1 and ARM Cortex-A7x1 Core Tile.
FVP_VE_Cortex_A17x1_A7x1.coretile.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_VE_Cortex_A17x1_A7x1.coretile.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.coretile.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster0	Cluster_ARM_Cortex-A17	ARM Cortex-A17 Cluster CT model.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster0.cpu0	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster0.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster1	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster1.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster1.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1_A7x1.coretile.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x1_A7x1.coretile.dualclustersystemconfigurationblock	*DualClusterSystemConfigurationBlock*	Dual Cluster System Configuration Block.
FVP_VE_Cortex_A17x1_A7x1.coretile.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A17x1_A7x1.coretile.v7_vgic	*v7_VGIC*	System VGIC architecture version v7.
FVP_VE_Cortex_A17x1_A7x1.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1_A7x1.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1_A7x1.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A17x1_A7x1.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A17x1_A7x1.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1_A7x1.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x1_A7x1.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A17x1_A7x1.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A17x1_A7x1.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x1_A7x1.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x1_A7x1.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x1_A7x1.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x1_A7x1.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x1_A7x1.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A17x1_A7x1.motherboard.mmc	*MMC*	Generic Multimedia Card.

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x1_A7x1.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A17x1_A7x1.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x1_A7x1.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x1_A7x1.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A17x1_A7x1.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A17x1_A7x1.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A17x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x1_A7x1.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x1_A7x1.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x1_A7x1.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x1_A7x1.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A17x1_A7x1.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A17x1_A7x1.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.

Table 5-8 FVP_VE_Cortex-A17x1-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x1_A7x1.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A17x1_A7x1.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A17x1_A7x1.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x1_A7x1.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.9 FVP_VE_Cortex-A17x2

FVP_VE_Cortex-A17x2 contains the following instances:

Table 5-9 FVP_VE_Cortex-A17x2 instances

Name	Type	Description
FVP_VE_Cortex_A17x2	FVP_VE_Cortex_A17x2	Top level component of the Cortex_A17x2 Versatile Express inspired model.
FVP_VE_Cortex_A17x2.cluster	Cluster_ARM_Cortex-A17	ARM Cortex-A17 Cluster CT model.
FVP_VE_Cortex_A17x2.cluster.cpu0	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x2.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x2.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x2.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x2.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x2.cluster.cpu1	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x2.cluster.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x2.cluster.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x2.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x2.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x2.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x2.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A17x2.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-9 FVP_VE_Cortex-A17x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x2.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x2.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x2.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A17x2.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A17x2.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x2.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x2.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x2.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x2.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A17x2.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x2.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x2.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x2.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-9 FVP_VE_Cortex-A17x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x2.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x2.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A17x2.gic400	*GIC_400*	GIC-400 Generic Interrupt Controller.
FVP_VE_Cortex_A17x2.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A17x2.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A17x2.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x2.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x2.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x2.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x2.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.

Table 5-9 FVP_VE_Cortex-A17x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x2.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x2.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A17x2.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A17x2.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x2.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.

Table 5-9 FVP_VE_Cortex-A17x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x2.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x2.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x2.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A17x2.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A17x2.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x2.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x2.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x2.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x2.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A17x2.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).

Table 5-9 FVP_VE_Cortex-A17x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x2.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A17x2.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A17x2.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A17x2.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A17x2.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x2.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).

Table 5-9 FVP_VE_Cortex-A17x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x2.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x2.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x2.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x2.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A17x2.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A17x2.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A17x2.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x2.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x2.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 5-9 FVP_VE_Cortex-A17x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x2.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x2.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A17x2.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A17x2.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A17x2.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A17x2.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A17x2.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x2.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.10 FVP_VE_Cortex-A17x4

FVP_VE_Cortex-A17x4 contains the following instances:

Table 5-10 FVP_VE_Cortex-A17x4 instances

Name	Type	Description
FVP_VE_Cortex_A17x4	FVP_VE_Cortex_A17x4	Top level component of the Cortex_A17x4 Versatile Express inspired model.
FVP_VE_Cortex_A17x4.cluster	Cluster_ARM_Cortex-A17	ARM Cortex-A17 Cluster CT model.
FVP_VE_Cortex_A17x4.cluster.cpu0	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.cpu1	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4.cluster.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.cpu2	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4.cluster.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu2.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.cpu3	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4.cluster.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu3.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4.cluster.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A17x4.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A17x4.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x4.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x4.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x4.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A17x4.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x4.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x4.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A17x4.gic400	*GIC_400*	GIC-400 Generic Interrupt Controller.
FVP_VE_Cortex_A17x4.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A17x4.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A17x4.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x4.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x4.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A17x4.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A17x4.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x4.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x4.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x4.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x4.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A17x4.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A17x4.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x4.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x4.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x4.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A17x4.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A17x4.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A17x4.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A17x4.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A17x4.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x4.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A17x4.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x4.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x4.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A17x4.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A17x4.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A17x4.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A17x4.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A17x4.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A17x4.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.

Table 5-10 FVP_VE_Cortex-A17x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A17x4.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.11 FVP_VE_Cortex-A17x4-A7x4

FVP_VE_Cortex-A17x4-A7x4 contains the following instances:

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4	FVP_VE_Cortex_A17x4_A7x4	Top level component of the Cortex A17x4 A7x4 Versatile Express inspired model.
FVP_VE_Cortex_A17x4_A7x4.coretile	ARM_Cortex_A17x4_A7x4_CT	Dual cluster ARM Cortex-A17x4 and ARM Cortex-A7x4 Core Tile.
FVP_VE_Cortex_A17x4_A7x4.coretile.cci400	*CCI400*	Cache Coherent Interconnect for AXI4 ACE.
FVP_VE_Cortex_A17x4_A7x4.coretile.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.coretile.clockdivider1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0	Cluster_ARM_Cortex-A17	ARM Cortex-A17 Cluster CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu0	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu1	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu1.l1icache	*PVCache*	PV Cache.

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu2	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu2.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu3	ARM_Cortex-A17	ARM Cortex-A17 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu3.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster0.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu1	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu1.l1dcache	*PVCache*	PV Cache.

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu2	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu2.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu3	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu3.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.cluster1.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A17x4_A7x4.coretile.dualclustersystemconfigurationblock	*DualClusterSystemConfigurationBlock*	Dual Cluster System Configuration Block.
FVP_VE_Cortex_A17x4_A7x4.coretile.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A17x4_A7x4.coretile.v7_vgic	*v7_VGIC*	System VGIC architecture version v7.
FVP_VE_Cortex_A17x4_A7x4.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A17x4_A7x4.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x4_A7x4.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x4_A7x4.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A17x4_A7x4.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A17x4_A7x4.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4_A7x4.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A17x4_A7x4.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A17x4_A7x4.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A17x4_A7x4.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x4_A7x4.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A17x4_A7x4.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x4_A7x4.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A17x4_A7x4.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A17x4_A7x4.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A17x4_A7x4.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A17x4_A7x4.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x4_A7x4.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A17x4_A7x4.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A17x4_A7x4.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A17x4_A7x4.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A17x4_A7x4.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4_A7x4.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4_A7x4.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4_A7x4.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A17x4_A7x4.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A17x4_A7x4.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A17x4_A7x4.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A17x4_A7x4.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A17x4_A7x4.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-11 FVP_VE_Cortex-A17x4-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A17x4_A7x4.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A17x4_A7x4.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.12 FVP_VE_Cortex-A5x1

FVP_VE_Cortex-A5x1 contains the following instances:

Table 5-12 FVP_VE_Cortex-A5x1 instances

Name	Type	Description
FVP_VE_Cortex_A5x1	FVP_VE_Cortex_A5x1	Top level component of the Cortex-A5x1 Versatile Express inspired model.
FVP_VE_Cortex_A5x1.cluster	Cluster_ARM_Cortex-A5MP	ARM CORTEXA5MP Cluster CT model.
FVP_VE_Cortex_A5x1.cluster.cpu0	ARM_Cortex-A5MP	ARM CORTEXA5MP CT model.
FVP_VE_Cortex_A5x1.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x1.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x1.cluster.cpu0.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A5x1.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A5x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x1.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x1.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.

Table 5-12 FVP_VE_Cortex-A5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x1.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A5x1.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x1.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x1.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A5x1.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A5x1.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A5x1.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x1.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x1.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x1.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x1.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A5x1.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.

Table 5-12 FVP_VE_Cortex-A5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x1.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A5x1.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x1.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x1.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A5x1.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x1.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x1.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A5x1.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-12 FVP_VE_Cortex-A5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x1.motherboard.clock 24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A5x1.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.motherboard.flash 0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x1.motherboard.flash 1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x1.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x1.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x1.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A5x1.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A5x1.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-12 FVP_VE_Cortex-A5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x1.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x1.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x1.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x1.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A5x1.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A5x1.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A5x1.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-12 FVP_VE_Cortex-A5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x1.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A5x1.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A5x1.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A5x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A5x1.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A5x1.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A5x1.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A5x1.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-12 FVP_VE_Cortex-A5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x1.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A5x1.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A5x1.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A5x1.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x1.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x1.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x1.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x1.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A5x1.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A5x1.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A5x1.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.

Table 5-12 FVP_VE_Cortex-A5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x1.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A5x1.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x1.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x1.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.13 FVP_VE_Cortex-A5x2

FVP_VE_Cortex-A5x2 contains the following instances:

Table 5-13 FVP_VE_Cortex-A5x2 instances

Name	Type	Description
FVP_VE_Cortex_A5x2	FVP_VE_Cortex_A5x2	Top level component of the Cortex-A5x2 Versatile Express inspired model.
FVP_VE_Cortex_A5x2.cluster	Cluster_ARM_Cortex-A5MP	ARM CORTEXA5MP Cluster CT model.
FVP_VE_Cortex_A5x2.cluster.cpu0	ARM_Cortex-A5MP	ARM CORTEXA5MP CT model.
FVP_VE_Cortex_A5x2.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x2.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x2.cluster.cpu0.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A5x2.cluster.cpu1	ARM_Cortex-A5MP	ARM CORTEXA5MP CT model.
FVP_VE_Cortex_A5x2.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x2.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x2.cluster.cpu1.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A5x2.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A5x2.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.

Table 5-13 FVP_VE_Cortex-A5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x2.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x2.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A5x2.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A5x2.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x2.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x2.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A5x2.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A5x2.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A5x2.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x2.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x2.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x2.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.

Table 5-13 FVP_VE_Cortex-A5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x2.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A5x2.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A5x2.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A5x2.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x2.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x2.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A5x2.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x2.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x2.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 5-13 FVP_VE_Cortex-A5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x2.motherboard.clock 100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.clock 24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A5x2.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x2.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x2.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x2.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-13 FVP_VE_Cortex-A5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x2.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A5x2.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A5x2.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x2.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x2.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x2.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x2.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A5x2.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A5x2.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-13 FVP_VE_Cortex-A5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x2.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A5x2.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A5x2.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A5x2.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A5x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A5x2.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A5x2.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 5-13 FVP_VE_Cortex-A5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x2.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A5x2.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A5x2.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A5x2.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A5x2.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x2.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x2.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x2.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 5-13 FVP_VE_Cortex-A5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x2.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A5x2.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A5x2.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A5x2.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A5x2.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A5x2.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x2.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x2.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.14 FVP_VE_Cortex-A5x4

FVP_VE_Cortex-A5x4 contains the following instances:

Table 5-14 FVP_VE_Cortex-A5x4 instances

Name	Type	Description
FVP_VE_Cortex_A5x4	FVP_VE_Cortex_A5x4	Top level component of the Cortex-A5x4 Versatile Express inspired model.
FVP_VE_Cortex_A5x4.cluster	Cluster_ARM_Cortex-A5MP	ARM CORTEXA5MP Cluster CT model.
FVP_VE_Cortex_A5x4.cluster.cpu0	ARM_Cortex-A5MP	ARM CORTEXA5MP CT model.
FVP_VE_Cortex_A5x4.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu0.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A5x4.cluster.cpu1	ARM_Cortex-A5MP	ARM CORTEXA5MP CT model.
FVP_VE_Cortex_A5x4.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu1.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A5x4.cluster.cpu2	ARM_Cortex-A5MP	ARM CORTEXA5MP CT model.
FVP_VE_Cortex_A5x4.cluster.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu2.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A5x4.cluster.cpu3	ARM_Cortex-A5MP	ARM CORTEXA5MP CT model.
FVP_VE_Cortex_A5x4.cluster.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A5x4.cluster.cpu3.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A5x4.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x4.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x4.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A5x4.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A5x4.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x4.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A5x4.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A5x4.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A5x4.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x4.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x4.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x4.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A5x4.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A5x4.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A5x4.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A5x4.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x4.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x4.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A5x4.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x4.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A5x4.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A5x4.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A5x4.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x4.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A5x4.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x4.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A5x4.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A5x4.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A5x4.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x4.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x4.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x4.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A5x4.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A5x4.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A5x4.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A5x4.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A5x4.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A5x4.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A5x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A5x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A5x4.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A5x4.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A5x4.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A5x4.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A5x4.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A5x4.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x4.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x4.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x4.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A5x4.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A5x4.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A5x4.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A5x4.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A5x4.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-14 FVP_VE_Cortex-A5x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A5x4.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A5x4.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A5x4.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.15 FVP_VE_Cortex-A7x1

FVP_VE_Cortex-A7x1 contains the following instances:

Table 5-15 FVP_VE_Cortex-A7x1 instances

Name	Type	Description
FVP_VE_Cortex_A7x1	FVP_VE_Cortex_A7x1	Top level component of the Cortex_A7x1 Versatile Express inspired model.
FVP_VE_Cortex_A7x1.cluster	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A7x1.cluster.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A7x1.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x1.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x1.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x1.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x1.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x1.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A7x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x1.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x1.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.

Table 5-15 FVP_VE_Cortex-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A7x1.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A7x1.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x1.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x1.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A7x1.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A7x1.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A7x1.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x1.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x1.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A7x1.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x1.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).

Table 5-15 FVP_VE_Cortex-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x1.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A7x1.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A7x1.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A7x1.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x1.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x1.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A7x1.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x1.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x1.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 5-15 FVP_VE_Cortex-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x1.motherboard.clock 100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.clock 24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A7x1.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x1.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x1.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A7x1.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-15 FVP_VE_Cortex-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x1.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A7x1.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A7x1.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x1.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x1.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x1.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x1.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A7x1.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A7x1.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-15 FVP_VE_Cortex-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x1.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A7x1.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A7x1.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A7x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A7x1.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A7x1.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 5-15 FVP_VE_Cortex-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x1.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A7x1.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x1.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A7x1.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A7x1.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A7x1.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x1.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x1.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x1.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 5-15 FVP_VE_Cortex-A7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x1.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A7x1.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A7x1.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A7x1.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A7x1.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A7x1.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x1.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.16 FVP_VE_Cortex-A7x2

FVP_VE_Cortex-A7x2 contains the following instances:

Table 5-16 FVP_VE_Cortex-A7x2 instances

Name	Type	Description
FVP_VE_Cortex_A7x2	FVP_VE_Cortex_A7x2	Top level component of the Cortex_A7x2 Versatile Express inspired model.
FVP_VE_Cortex_A7x2.cluster	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A7x2.cluster.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A7x2.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x2.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x2.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x2.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x2.cluster.cpu1	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A7x2.cluster.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x2.cluster.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x2.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x2.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x2.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x2.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A7x2.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-16 FVP_VE_Cortex-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x2.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x2.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x2.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A7x2.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A7x2.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x2.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x2.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A7x2.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A7x2.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A7x2.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x2.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x2.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A7x2.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-16 FVP_VE_Cortex-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x2.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x2.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A7x2.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A7x2.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A7x2.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A7x2.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x2.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x2.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A7x2.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x2.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 5-16 FVP_VE_Cortex-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x2.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A7x2.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A7x2.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x2.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x2.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-16 FVP_VE_Cortex-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x2.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A7x2.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A7x2.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A7x2.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x2.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x2.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x2.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x2.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A7x2.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A7x2.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-16 FVP_VE_Cortex-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x2.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A7x2.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A7x2.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A7x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A7x2.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A7x2.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 5-16 FVP_VE_Cortex-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x2.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A7x2.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x2.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A7x2.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A7x2.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A7x2.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x2.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x2.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x2.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 5-16 FVP_VE_Cortex-A7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x2.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A7x2.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A7x2.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A7x2.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A7x2.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A7x2.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x2.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.17 FVP_VE_Cortex-A7x4

FVP_VE_Cortex-A7x4 contains the following instances:

Table 5-17 FVP_VE_Cortex-A7x4 instances

Name	Type	Description
FVP_VE_Cortex_A7x4	FVP_VE_Cortex_A7x4	Top level component of the Cortex_A7x3 Versatile Express inspired model.
FVP_VE_Cortex_A7x4.cluster	Cluster_ARM_Cortex-A7	ARM Cortex-A7 Cluster CT model.
FVP_VE_Cortex_A7x4.cluster.cpu0	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A7x4.cluster.cpu0.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu0.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.cpu1	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A7x4.cluster.cpu1.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu1.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.cpu2	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A7x4.cluster.cpu2.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu2.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.cpu3	ARM_Cortex-A7	ARM Cortex-A7 CT model.
FVP_VE_Cortex_A7x4.cluster.cpu3.dtlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu3.itlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A7x4.cluster.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.cluster.l2_cache	*PVCache*	PV Cache.
FVP_VE_Cortex_A7x4.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x4.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x4.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A7x4.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A7x4.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x4.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A7x4.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A7x4.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A7x4.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x4.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x4.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A7x4.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A7x4.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A7x4.globalcounter	*MemoryMappedCounterModule*	Memory Mapped Counter Module for Generic Timers.
FVP_VE_Cortex_A7x4.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A7x4.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A7x4.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x4.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x4.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A7x4.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x4.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A7x4.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A7x4.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A7x4.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.motherboard.flash 0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x4.motherboard.flash 1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A7x4.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A7x4.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A7x4.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A7x4.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A7x4.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x4.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x4.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x4.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A7x4.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A7x4.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A7x4.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A7x4.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A7x4.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A7x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A7x4.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A7x4.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A7x4.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A7x4.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A7x4.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A7x4.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A7x4.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x4.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x4.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x4.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A7x4.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A7x4.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A7x4.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A7x4.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A7x4.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-17 FVP_VE_Cortex-A7x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A7x4.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A7x4.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.

5.18 FVP_VE_Cortex-A9x1

FVP_VE_Cortex-A9x1 contains the following instances:

Table 5-18 FVP_VE_Cortex-A9x1 instances

Name	Type	Description
FVP_VE_Cortex_A9x1	FVP_VE_Cortex_A9x1	Top level component of the Cortex-A9x1 Versatile Express inspired model.
FVP_VE_Cortex_A9x1.cluster	Cluster_ARM_Cortex-A9MP	ARM CORTEXA9MP Cluster CT model.
FVP_VE_Cortex_A9x1.cluster.cpu0	ARM_Cortex-A9MP	ARM CORTEXA9MP CT model.
FVP_VE_Cortex_A9x1.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x1.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x1.cluster.cpu0.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A9x1.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A9x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x1.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x1.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.

Table 5-18 FVP_VE_Cortex-A9x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x1.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A9x1.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x1.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x1.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A9x1.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A9x1.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A9x1.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x1.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x1.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x1.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x1.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A9x1.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.

Table 5-18 FVP_VE_Cortex-A9x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x1.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A9x1.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x1.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x1.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A9x1.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x1.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x1.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A9x1.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-18 FVP_VE_Cortex-A9x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x1.motherboard.clock 24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A9x1.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.motherboard.flash 0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x1.motherboard.flash 1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x1.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x1.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x1.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A9x1.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A9x1.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-18 FVP_VE_Cortex-A9x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x1.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x1.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x1.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x1.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A9x1.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A9x1.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A9x1.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-18 FVP_VE_Cortex-A9x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x1.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A9x1.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A9x1.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A9x1.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A9x1.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A9x1.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A9x1.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A9x1.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-18 FVP_VE_Cortex-A9x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x1.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A9x1.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A9x1.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A9x1.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x1.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x1.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x1.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x1.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A9x1.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A9x1.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A9x1.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.

Table 5-18 FVP_VE_Cortex-A9x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x1.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A9x1.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x1.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x1.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.19 FVP_VE_Cortex-A9x2

FVP_VE_Cortex-A9x2 contains the following instances:

Table 5-19 FVP_VE_Cortex-A9x2 instances

Name	Type	Description
FVP_VE_Cortex_A9x2	FVP_VE_Cortex_A9x2	Top level component of the Cortex-A9x2 Versatile Express inspired model.
FVP_VE_Cortex_A9x2.cluster	Cluster_ARM_Cortex-A9MP	ARM CORTEXA9MP Cluster CT model.
FVP_VE_Cortex_A9x2.cluster.cpu0	ARM_Cortex-A9MP	ARM CORTEXA9MP CT model.
FVP_VE_Cortex_A9x2.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x2.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x2.cluster.cpu0.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A9x2.cluster.cpu1	ARM_Cortex-A9MP	ARM CORTEXA9MP CT model.
FVP_VE_Cortex_A9x2.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x2.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x2.cluster.cpu1.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A9x2.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.
FVP_VE_Cortex_A9x2.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.

Table 5-19 FVP_VE_Cortex-A9x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x2.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x2.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A9x2.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A9x2.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x2.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x2.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A9x2.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A9x2.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A9x2.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x2.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x2.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x2.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.

Table 5-19 FVP_VE_Cortex-A9x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x2.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A9x2.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A9x2.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A9x2.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x2.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x2.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A9x2.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x2.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x2.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 5-19 FVP_VE_Cortex-A9x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x2.motherboard.clock 100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.clock 24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A9x2.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x2.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x2.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x2.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-19 FVP_VE_Cortex-A9x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x2.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A9x2.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A9x2.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x2.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x2.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x2.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x2.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A9x2.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A9x2.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-19 FVP_VE_Cortex-A9x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x2.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A9x2.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_A9x2.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A9x2.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A9x2.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A9x2.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A9x2.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.

Table 5-19 FVP_VE_Cortex-A9x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x2.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A9x2.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A9x2.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A9x2.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_A9x2.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x2.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x2.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x2.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.

Table 5-19 FVP_VE_Cortex-A9x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x2.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A9x2.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A9x2.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A9x2.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A9x2.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_A9x2.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x2.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x2.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.20 FVP_VE_Cortex-A9x4

FVP_VE_Cortex-A9x4 contains the following instances:

Table 5-20 FVP_VE_Cortex-A9x4 instances

Name	Type	Description
FVP_VE_Cortex_A9x4	FVP_VE_Cortex_A9x4	Top level component of the Cortex-A9x4 Versatile Express inspired model.
FVP_VE_Cortex_A9x4.cluster	Cluster_ARM_Cortex-A9MP	ARM CORTEXA9MP Cluster CT model.
FVP_VE_Cortex_A9x4.cluster.cpu0	ARM_Cortex-A9MP	ARM CORTEXA9MP CT model.
FVP_VE_Cortex_A9x4.cluster.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu0.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A9x4.cluster.cpu1	ARM_Cortex-A9MP	ARM CORTEXA9MP CT model.
FVP_VE_Cortex_A9x4.cluster.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu1.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A9x4.cluster.cpu2	ARM_Cortex-A9MP	ARM CORTEXA9MP CT model.
FVP_VE_Cortex_A9x4.cluster.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu2.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A9x4.cluster.cpu3	ARM_Cortex-A9MP	ARM CORTEXA9MP CT model.
FVP_VE_Cortex_A9x4.cluster.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_A9x4.cluster.cpu3.utlb	TlbCadi	TLB - instruction, data or unified.
FVP_VE_Cortex_A9x4.daughterboard	VEDaughterBoard	Daughtercard, inspired by the ARM Versatile Express development platform.

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.daughterboard.dmc	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.daughterboard.dmc_phy	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.daughterboard.dram_aliased	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x4.daughterboard.dram_limit_4	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x4.daughterboard.dram_limit_8	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_A9x4.daughterboard.hdlcd	*PL370_HDLCD*	ARM PrimeCell HD Color LCD Controller (Nominal Designation PL370).
FVP_VE_Cortex_A9x4.daughterboard.hdlcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.daughterboard.hdlcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x4.daughterboard.hdlcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A9x4.daughterboard.hdlcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A9x4.daughterboard.introuter	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_A9x4.daughterboard.nonsecure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x4.daughterboard.secureDRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.daughterboard.secureRO	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x4.daughterboard.secureROloader	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x4.daughterboard.secureSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.daughterboard.secure_region	*TZSwitch*	Allow TrustZone secure/normal bus signals to be routed separately.
FVP_VE_Cortex_A9x4.daughterboard.sram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.daughterboard.vedcc	VEDCC	Daughterboard Configuration Control (DCC).
FVP_VE_Cortex_A9x4.motherboard	VEMotherBoard	Model inspired by the ARM Versatile Express Motherboard.
FVP_VE_Cortex_A9x4.motherboard.Timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A9x4.motherboard.Timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.motherboard.Timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.Timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x4.motherboard.Timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x4.motherboard.Timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_A9x4.motherboard.Timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.Timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.Timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x4.motherboard.Timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_A9x4.motherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_A9x4.motherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.motherboard.clock 35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_A9x4.motherboard.dummy_local_dap_rom	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.motherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.motherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.motherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x4.motherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_A9x4.motherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x4.motherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_A9x4.motherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_A9x4.motherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_A9x4.motherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x4.motherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.motherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x4.motherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x4.motherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_A9x4.motherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_A9x4.motherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_A9x4.motherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A9x4.motherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_A9x4.motherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.motherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_A9x4.motherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_A9x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_A9x4.motherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_A9x4.motherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_A9x4.motherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A9x4.motherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_A9x4.motherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.motherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_A9x4.motherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_A9x4.motherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.motherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x4.motherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x4.motherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x4.motherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_A9x4.motherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_A9x4.motherboard.virtioblockdevice	*VirtioBlockDevice*	virtio block device.
FVP_VE_Cortex_A9x4.motherboard.virtiop9device	*VirtioP9Device*	virtio P9 server.
FVP_VE_Cortex_A9x4.motherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_A9x4.motherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-20 FVP_VE_Cortex-A9x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_A9x4.motherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_A9x4.motherboard.vram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_A9x4.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.21 FVP_VE_Cortex-R4

FVP_VE_Cortex-R4 contains the following instances:

Table 5-21 FVP_VE_Cortex-R4 instances

Name	Type	Description
FVP_VE_Cortex_R4	FVP_VE_Cortex_R4	Top level component of the Cortex_R4 Versatile Express inspired model.
FVP_VE_Cortex_R4.daughterboard	VEDaughterBoardCortex_R4	Cortex-R4 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R4.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.daughterboard.core	ARM_Cortex-R4	ARM CORTEXR4 CT model.
FVP_VE_Cortex_R4.daughterboard.core.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R4.daughterboard.core.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R4.daughterboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R4.daughterboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R4.daughterboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-21 FVP_VE_Cortex-R4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R4.daughterboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R4.daughterboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R4.daughterboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R4.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R4.daughterboard.pl390_gic	*PL390_GIC*	Generic Interrupt Controller (PL390).
FVP_VE_Cortex_R4.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R4.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.
FVP_VE_Cortex_R4.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_R4.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-21 FVP_VE_Cortex-R4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R4.vemotherboard.clock 50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.clock CLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R4.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R4.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R4.vemotherboard.flash 0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R4.vemotherboard.flash 1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R4.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R4.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R4.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_R4.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R4.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R4.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R4.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-21 FVP_VE_Cortex-R4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R4.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R4.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R4.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R4.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_R4.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_R4.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R4.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R4.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).

Table 5-21 FVP_VE_Cortex-R4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R4.vemotherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R4.vemotherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R4.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R4.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R4.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R4.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_R4.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R4.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R4.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R4.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R4.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R4.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-21 FVP_VE_Cortex-R4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R4.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R4.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R4.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R4.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R4.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R4.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R4.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-21 FVP_VE_Cortex-R4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R4.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R4.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R4.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R4.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R4.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R4.vemotherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_R4.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R4.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_R4.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-21 FVP_VE_Cortex-R4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R4.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R4.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.22 FVP_VE_Cortex-R5x1

FVP_VE_Cortex-R5x1 contains the following instances:

Table 5-22 FVP_VE_Cortex-R5x1 instances

Name	Type	Description
FVP_VE_Cortex_R5x1	FVP_VE_Cortex_R5x1	Top level component of the Cortex_R5x1 Versatile Express inspired model.
FVP_VE_Cortex_R5x1.daughterboard	VEDaughterBoardCortex_R5x1	Cortex-R5x1 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R5x1.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.daughterboard.core	Cluster_ARM_Cortex-R5	ARM CORTEXR5 Cluster CT model.
FVP_VE_Cortex_R5x1.daughterboard.core.cpu0	ARM_Cortex-R5	ARM CORTEXR5 CT model.
FVP_VE_Cortex_R5x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R5x1.daughterboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R5x1.daughterboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R5x1.daughterboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-22 FVP_VE_Cortex-R5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x1.daughterboard.pl111_clcd.pl111x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R5x1.daughterboard.pl111_clcd.pl111x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R5x1.daughterboard.pl111_clcd.pl111x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R5x1.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R5x1.daughterboard.pl390_gic	*PL390_GIC*	Generic Interrupt Controller (PL390).
FVP_VE_Cortex_R5x1.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R5x1.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.
FVP_VE_Cortex_R5x1.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_R5x1.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-22 FVP_VE_Cortex-R5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x1.vemotherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R5x1.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x1.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x1.vemotherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R5x1.vemotherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R5x1.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R5x1.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R5x1.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_R5x1.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-22 FVP_VE_Cortex-R5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x1.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_R5x1.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_R5x1.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R5x1.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-22 FVP_VE_Cortex-R5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x1.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R5x1.vemotherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R5x1.vemotherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R5x1.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R5x1.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R5x1.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R5x1.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_R5x1.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R5x1.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R5x1.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-22 FVP_VE_Cortex-R5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x1.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R5x1.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R5x1.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_R5x1.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x1.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x1.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x1.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x1.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x1.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).

Table 5-22 FVP_VE_Cortex-R5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x1.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x1.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x1.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R5x1.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x1.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x1.vemotherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_R5x1.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x1.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_R5x1.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-22 FVP_VE_Cortex-R5x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x1.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x1.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.23 FVP_VE_Cortex-R5x2

FVP_VE_Cortex-R5x2 contains the following instances:

Table 5-23 FVP_VE_Cortex-R5x2 instances

Name	Type	Description
FVP_VE_Cortex_R5x2	FVP_VE_Cortex_R5x2	Top level component of the Cortex_R5x2 Versatile Express inspired model.
FVP_VE_Cortex_R5x2.daughterboard	VEDaughterBoardCortex_R5x2	Cortex-R5x2 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R5x2.daughterboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.daughterboard.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.daughterboard.core	Cluster_ARM_Cortex-R5	ARM CORTEXR5 Cluster CT model.
FVP_VE_Cortex_R5x2.daughterboard.core.cpu0	ARM_Cortex-R5	ARM CORTEXR5 CT model.
FVP_VE_Cortex_R5x2.daughterboard.core.cpu1	ARM_Cortex-R5	ARM CORTEXR5 CT model.
FVP_VE_Cortex_R5x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R5x2.daughterboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R5x2.daughterboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R5x2.daughterboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 5-23 FVP_VE_Cortex-R5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x2.daughterboard.pl111_clcd.pl111x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R5x2.daughterboard.pl111_clcd.pl111x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R5x2.daughterboard.pl111_clcd.pl111x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R5x2.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R5x2.daughterboard.pl390_gic	*PL390_GIC*	Generic Interrupt Controller (PL390).
FVP_VE_Cortex_R5x2.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R5x2.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.
FVP_VE_Cortex_R5x2.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_R5x2.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-23 FVP_VE_Cortex-R5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x2.vemotherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R5x2.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x2.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x2.vemotherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R5x2.vemotherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R5x2.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R5x2.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R5x2.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_R5x2.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-23 FVP_VE_Cortex-R5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R5x2.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_R5x2.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_R5x2.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R5x2.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).

Table 5-23 FVP_VE_Cortex-R5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x2.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R5x2.vemotherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R5x2.vemotherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R5x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R5x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R5x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R5x2.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_R5x2.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R5x2.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R5x2.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-23 FVP_VE_Cortex-R5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x2.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R5x2.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R5x2.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_R5x2.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x2.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x2.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x2.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x2.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R5x2.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).

Table 5-23 FVP_VE_Cortex-R5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x2.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x2.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x2.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R5x2.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x2.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R5x2.vemotherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_R5x2.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R5x2.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_R5x2.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-23 FVP_VE_Cortex-R5x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R5x2.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R5x2.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.24 FVP_VE_Cortex-R7x1

FVP_VE_Cortex-R7x1 contains the following instances:

Table 5-24 FVP_VE_Cortex-R7x1 instances

Name	Type	Description
FVP_VE_Cortex_R7x1	FVP_VE_Cortex_R7x1	Top level component of the Cortex_R7x1 Versatile Express inspired model.
FVP_VE_Cortex_R7x1.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.daughterboard	VEDaughterBoardCortex_R7x1	Cortex_R7x1 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R7x1.daughterboard.core	Cluster_ARM_Cortex-R7	ARM CORTEXR7 Cluster CT model.
FVP_VE_Cortex_R7x1.daughterboard.core.cpu0	ARM_Cortex-R7	ARM CORTEXR7 CT model.
FVP_VE_Cortex_R7x1.daughterboard.core.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R7x1.daughterboard.core.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R7x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R7x1.daughterboard.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R7x1.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R7x1.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.
FVP_VE_Cortex_R7x1.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 5-24 FVP_VE_Cortex-R7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x1.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R7x1.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x1.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x1.vemotherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R7x1.vemotherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R7x1.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R7x1.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R7x1.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.

Table 5-24 FVP_VE_Cortex-R7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x1.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x1.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_R7x1.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).

Table 5-24 FVP_VE_Cortex-R7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x1.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R7x1.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R7x1.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R7x1.vemotherboard.pl111_clcd.pl111x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R7x1.vemotherboard.pl111_clcd.pl111x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R7x1.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R7x1.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R7x1.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R7x1.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).

Table 5-24 FVP_VE_Cortex-R7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x1.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R7x1.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R7x1.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x1.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R7x1.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R7x1.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_R7x1.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x1.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x1.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 5-24 FVP_VE_Cortex-R7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x1.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x1.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x1.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R7x1.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x1.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x1.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R7x1.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x1.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x1.vemotherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_R7x1.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x1.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.

Table 5-24 FVP_VE_Cortex-R7x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x1.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_R7x1.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x1.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.25 FVP_VE_Cortex-R7x2

FVP_VE_Cortex-R7x2 contains the following instances:

Table 5-25 FVP_VE_Cortex-R7x2 instances

Name	Type	Description
FVP_VE_Cortex_R7x2	FVP_VE_Cortex_R7x2	Top level component of the Cortex_R7x2 Versatile Express inspired model.
FVP_VE_Cortex_R7x2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.daughterboard	VEDaughterBoardCortex_R7x2	Cortex_R7x2 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R7x2.daughterboard.core	Cluster_ARM_Cortex-R7	ARM CORTEXR7 Cluster CT model.
FVP_VE_Cortex_R7x2.daughterboard.core.cpu0	ARM_Cortex-R7	ARM CORTEXR7 CT model.
FVP_VE_Cortex_R7x2.daughterboard.core.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R7x2.daughterboard.core.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R7x2.daughterboard.core.cpu1	ARM_Cortex-R7	ARM CORTEXR7 CT model.
FVP_VE_Cortex_R7x2.daughterboard.core.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R7x2.daughterboard.core.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R7x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R7x2.daughterboard.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R7x2.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R7x2.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.

Table 5-25 FVP_VE_Cortex-R7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x2.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_R7x2.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R7x2.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x2.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x2.vemotherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R7x2.vemotherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R7x2.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R7x2.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-25 FVP_VE_Cortex-R7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x2.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_R7x2.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R7x2.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).

Table 5-25 FVP_VE_Cortex-R7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x2.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_R7x2.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R7x2.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R7x2.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R7x2.vemotherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R7x2.vemotherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R7x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R7x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R7x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 5-25 FVP_VE_Cortex-R7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x2.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_R7x2.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R7x2.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R7x2.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x2.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R7x2.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R7x2.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_R7x2.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x2.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.

Table 5-25 FVP_VE_Cortex-R7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x2.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x2.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x2.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R7x2.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R7x2.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x2.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x2.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R7x2.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x2.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R7x2.vemotherboard.ve_sysregs	VE_SysRegs	-

Table 5-25 FVP_VE_Cortex-R7x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R7x2.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R7x2.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_R7x2.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_R7x2.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R7x2.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.26 FVP_VE_Cortex-R8x1

FVP_VE_Cortex-R8x1 contains the following instances:

Table 5-26 FVP_VE_Cortex-R8x1 instances

Name	Type	Description
FVP_VE_Cortex_R8x1	FVP_VE_Cortex_R8x1	Top level component of the Cortex_R8x1 Versatile Express inspired model.
FVP_VE_Cortex_R8x1.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.daughterboard	VEDaughterBoardCortex_R8x1	Cortex_R8x1 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R8x1.daughterboard.core	Cluster_ARM_Cortex-R8	ARM CORTEXR8 Cluster CT model.
FVP_VE_Cortex_R8x1.daughterboard.core.cpu0	ARM_Cortex-R8	ARM CORTEXR8 CT model.
FVP_VE_Cortex_R8x1.daughterboard.core.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x1.daughterboard.core.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x1.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x1.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R8x1.daughterboard.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R8x1.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R8x1.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.
FVP_VE_Cortex_R8x1.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.

Table 5-26 FVP_VE_Cortex-R8x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x1.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R8x1.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x1.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x1.vemotherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R8x1.vemotherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R8x1.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R8x1.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R8x1.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.

Table 5-26 FVP_VE_Cortex-R8x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x1.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x1.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_R8x1.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).

Table 5-26 FVP_VE_Cortex-R8x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x1.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R8x1.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R8x1.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R8x1.vemotherboard.pl111_clcd.pl111x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R8x1.vemotherboard.pl111_clcd.pl111x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R8x1.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R8x1.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R8x1.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R8x1.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).

Table 5-26 FVP_VE_Cortex-R8x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x1.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R8x1.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R8x1.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x1.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R8x1.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R8x1.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_R8x1.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x1.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x1.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.

Table 5-26 FVP_VE_Cortex-R8x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x1.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x1.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x1.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R8x1.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x1.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x1.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R8x1.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x1.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x1.vemotherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_R8x1.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x1.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.

Table 5-26 FVP_VE_Cortex-R8x1 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x1.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_R8x1.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x1.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.27 FVP_VE_Cortex-R8x2

FVP_VE_Cortex-R8x2 contains the following instances:

Table 5-27 FVP_VE_Cortex-R8x2 instances

Name	Type	Description
FVP_VE_Cortex_R8x2	FVP_VE_Cortex_R8x2	Top level component of the Cortex_R8x2 Versatile Express inspired model.
FVP_VE_Cortex_R8x2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.daughterboard	VEDaughterBoardCortex_R8x2	Cortex_R8x2 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R8x2.daughterboard.core	Cluster_ARM_Cortex-R8	ARM CORTEXR8 Cluster CT model.
FVP_VE_Cortex_R8x2.daughterboard.core.cpu0	ARM_Cortex-R8	ARM CORTEXR8 CT model.
FVP_VE_Cortex_R8x2.daughterboard.core.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x2.daughterboard.core.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x2.daughterboard.core.cpu1	ARM_Cortex-R8	ARM CORTEXR8 CT model.
FVP_VE_Cortex_R8x2.daughterboard.core.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x2.daughterboard.core.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x2.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x2.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R8x2.daughterboard.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R8x2.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R8x2.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.

Table 5-27 FVP_VE_Cortex-R8x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x2.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_R8x2.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R8x2.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x2.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x2.vemotherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R8x2.vemotherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R8x2.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R8x2.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.

Table 5-27 FVP_VE_Cortex-R8x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x2.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_R8x2.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x2.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).

Table 5-27 FVP_VE_Cortex-R8x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x2.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_R8x2.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R8x2.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R8x2.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R8x2.vemotherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.
FVP_VE_Cortex_R8x2.vemotherboard.pl111_clcd.pl11x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R8x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R8x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R8x2.vemotherboard.pl111_clcd.pl11x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 5-27 FVP_VE_Cortex-R8x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x2.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_R8x2.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R8x2.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R8x2.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x2.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R8x2.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R8x2.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.
FVP_VE_Cortex_R8x2.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x2.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.

Table 5-27 FVP_VE_Cortex-R8x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x2.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x2.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x2.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x2.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R8x2.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x2.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x2.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R8x2.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x2.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x2.vemotherboard.ve_sysregs	VE_SysRegs	-

Table 5-27 FVP_VE_Cortex-R8x2 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x2.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x2.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_R8x2.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).
FVP_VE_Cortex_R8x2.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x2.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

5.28 FVP_VE_Cortex-R8x4

FVP_VE_Cortex-R8x4 contains the following instances:

Table 5-28 FVP_VE_Cortex-R8x4 instances

Name	Type	Description
FVP_VE_Cortex_R8x4	FVP_VE_Cortex_R8x4	Top level component of the Cortex_R8x4 Versatile Express inspired model.
FVP_VE_Cortex_R8x4.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.daughterboard	VEDaughterBoardCortex_R8x4	Cortex_R8x4 DaughterBoard for Versatile Express.
FVP_VE_Cortex_R8x4.daughterboard.core	Cluster_ARM_Cortex-R8	ARM CORTEXR8 Cluster CT model.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu0	ARM_Cortex-R8	ARM CORTEXR8 CT model.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu0.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu0.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu1	ARM_Cortex-R8	ARM CORTEXR8 CT model.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu1.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu1.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu2	ARM_Cortex-R8	ARM CORTEXR8 CT model.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu2.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu2.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu3	ARM_Cortex-R8	ARM CORTEXR8 CT model.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu3.l1dcache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.core.cpu3.l1icache	*PVCache*	PV Cache.
FVP_VE_Cortex_R8x4.daughterboard.dram	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 5-28 FVP_VE_Cortex-R8x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x4.daughterboard.exclusive_monitor	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_VE_Cortex_R8x4.daughterboard.periph_clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.daughterboard.pl310_l2cc	*PL310_L2CC*	ARM PrimeCell Level 2 Cache Controller (PL310).
FVP_VE_Cortex_R8x4.daughterboard.veinterruptmapper	VEInterruptMapper	Interrupt Mapping peripheral (non-cascaded).
FVP_VE_Cortex_R8x4.vemotherboard	VEMotherBoardR	Model inspired by the ARM Versatile Express Motherboard for R profile.
FVP_VE_Cortex_R8x4.vemotherboard.audioout	*AudioOut_SDL*	SDL based Audio Output for PL041_AACI.
FVP_VE_Cortex_R8x4.vemotherboard.clock100Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.clock24MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.clock35MHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.clockCLCD	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-28 FVP_VE_Cortex-R8x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x4.vemotherboard.dummy_CF	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_VE_Cortex_R8x4.vemotherboard.dummy_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x4.vemotherboard.dummy_usb	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x4.vemotherboard.flash0	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R8x4.vemotherboard.flash1	*IntelStrataFlashJ3*	Intel Strata Flash J3 LISA+ model.
FVP_VE_Cortex_R8x4.vemotherboard.flashloader0	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R8x4.vemotherboard.flashloader1	*FlashLoader*	A device that can preload a gzipped image into flash at startup.
FVP_VE_Cortex_R8x4.vemotherboard.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_VE_Cortex_R8x4.vemotherboard.mmc	*MMC*	Generic Multimedia Card.
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart0	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart1	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart2	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart3	*PL011_Uart*	ARM PrimeCell UART(PL011).

Table 5-28 FVP_VE_Cortex-R8x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart3.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart4	*PL011_Uart*	ARM PrimeCell UART(PL011).
FVP_VE_Cortex_R8x4.vemotherboard.pl011_uart4.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.pl031_rtc	*PL031_RTC*	ARM PrimeCell Real Time Clock(PL031).
FVP_VE_Cortex_R8x4.vemotherboard.pl041_aaci	*PL041_AACI*	ARM PrimeCell Advanced Audio CODEC Interface(PL041).
FVP_VE_Cortex_R8x4.vemotherboard.pl050_kmi0	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R8x4.vemotherboard.pl050_kmi0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.pl050_kmi1	*PL050_KMI*	ARM PrimeCell PS2 Keyboard/Mouse Interface(PL050).
FVP_VE_Cortex_R8x4.vemotherboard.pl050_kmi1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.pl111_clcd	*PL111_CLCD*	ARM PrimeCell Color LCD Controller(PL111).
FVP_VE_Cortex_R8x4.vemotherboard.pl111_clcd.pl11x_clcd	*PL11x_CLCD*	Internal component used by PL110 and PL111 CLCD controllers.

Table 5-28 FVP_VE_Cortex-R8x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x4.vemotherboard.pl111_clcd.pl111x_clcd.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R8x4.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_VE_Cortex_R8x4.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_VE_Cortex_R8x4.vemotherboard.pl111_clcd.pl111x_clcd.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_VE_Cortex_R8x4.vemotherboard.pl180_mci	*PL180_MCI*	ARM PrimeCell Multimedia Card Interface (PL180).
FVP_VE_Cortex_R8x4.vemotherboard.ps2keyboard	*PS2Keyboard*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R8x4.vemotherboard.ps2mouse	*PS2Mouse*	Interface component, which takes the keypress/release signals from the Visualisation component and translates them into clocked PS2Data signals which can be routed to a PL050_KMI component.
FVP_VE_Cortex_R8x4.vemotherboard.psram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x4.vemotherboard.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_VE_Cortex_R8x4.vemotherboard.sp805_wdog	*SP805_Watchdog*	ARM Watchdog Module(SP805).
FVP_VE_Cortex_R8x4.vemotherboard.sp810_sysctrl	*SP810_SysCtrl*	Only EB relevant functionalities are fully implemented.

Table 5-28 FVP_VE_Cortex-R8x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x4.vemotherboard.sp810_sysctrl.clkdiv_clk0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.sp810_sysctrl.clkdiv_clk1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.sp810_sysctrl.clkdiv_clk2	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.sp810_sysctrl.clkdiv_clk3	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.terminal_0	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x4.vemotherboard.terminal_1	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x4.vemotherboard.terminal_2	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x4.vemotherboard.terminal_3	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x4.vemotherboard.terminal_4	*TelnetTerminal*	Telnet terminal interface.
FVP_VE_Cortex_R8x4.vemotherboard.timer_0_1	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R8x4.vemotherboard.timer_0_1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 5-28 FVP_VE_Cortex-R8x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x4.vemotherboard.timer_0_1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.timer_0_1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x4.vemotherboard.timer_0_1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x4.vemotherboard.timer_2_3	*SP804_Timer*	ARM Dual-Timer Module(SP804).
FVP_VE_Cortex_R8x4.vemotherboard.timer_2_3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.timer_2_3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.timer_2_3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x4.vemotherboard.timer_2_3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_VE_Cortex_R8x4.vemotherboard.ve_sysregs	VE_SysRegs	-
FVP_VE_Cortex_R8x4.vemotherboard.video_ram	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_VE_Cortex_R8x4.vemotherboard.vis	VEVisualisation	Display window for VE using Visualisation library.
FVP_VE_Cortex_R8x4.vemotherboard.vis.recorder	*VisEventRecorder*	Event recorder component for visualisation component (allows to playback and record interactive GUI sessions).

Table 5-28 FVP_VE_Cortex-R8x4 instances (continued)

Name	Type	Description
FVP_VE_Cortex_R8x4.vemotherboard.vis.recorder.playbackDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_VE_Cortex_R8x4.vemotherboard.vis.recorder.recordingDivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Chapter 6
MPS2 Platform FVPs

This chapter lists the MPS2 Platform FVPs and the instances in them.

For the MPS2 memory maps, see *MPS2 - memory maps* in the Fast Models Reference Manual.

It contains the following sections:

- *6.1 FVP_MPS2_Cortex-M0* on page 6-993.
- *6.2 FVP_MPS2_Cortex-M0plus* on page 6-1007.
- *6.3 FVP_MPS2_Cortex-M23* on page 6-1022.
- *6.4 FVP_MPS2_Cortex-M3* on page 6-1037.
- *6.5 FVP_MPS2_Cortex-M33* on page 6-1051.
- *6.6 FVP_MPS2_Cortex-M35P* on page 6-1066.
- *6.7 FVP_MPS2_Cortex-M4* on page 6-1081.
- *6.8 FVP_MPS2_Cortex-M55* on page 6-1095.
- *6.9 FVP_MPS2_Cortex-M7* on page 6-1110.

6.1 FVP_MPS2_Cortex-M0

FVP_MPS2_Cortex-M0 contains the following instances:

Table 6-1 FVP_MPS2_Cortex-M0 instances

Name	Type	Description
FVP_MPS2_Cortex_M0	FVP_MPS2_Cortex_M0	-
FVP_MPS2_Cortex_M0.armcortexm0ct	ARM_Cortex-M0	ARM CORTEXM0 CT model.
FVP_MPS2_Cortex_M0.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M0.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M0.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M0.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M0.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M0.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M0.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M0.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M0.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M0.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M0.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M0.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M0.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M0.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M0.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M0.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M0.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M0.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M0.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-1 FVP_MPS2_Cortex-M0 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M0.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M0.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M0.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M0.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.2 FVP_MPS2_Cortex-M0plus

FVP_MPS2_Cortex-M0plus contains the following instances:

Table 6-2 FVP_MPS2_Cortex-M0plus instances

Name	Type	Description
FVP_MPS2_Cortex_M0plus	FVP_MPS2_Cortex_M0plus	-
FVP_MPS2_Cortex_M0plus.armcortexm0plusct	ARM_Cortex-M0+	ARM CORTEXM0+ CT model.
FVP_MPS2_Cortex_M0plus.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M0plus.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0plus.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0plus.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0plus.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M0plus.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M0plus.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M0plus.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M0plus.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M0plus.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.
FVP_MPS2_Cortex_M0plus.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M0plus.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M0plus.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M0plus.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M0plus.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-2 FVP_MPS2_Cortex-M0plus instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M0plus.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M0plus.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M0plus.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M0plus.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M0plus.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M0plus.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.3 FVP_MPS2_Cortex-M23

FVP_MPS2_Cortex-M23 contains the following instances:

Table 6-3 FVP_MPS2_Cortex-M23 instances

Name	Type	Description
FVP_MPS2_Cortex_M23	FVP_MPS2_Cortex_M23	-
FVP_MPS2_Cortex_M23.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.cpu0	ARM_Cortex-M23	ARM CORTEXM23 CT model.
FVP_MPS2_Cortex_M23.cpu1	ARM_Cortex-M23	ARM CORTEXM23 CT model.
FVP_MPS2_Cortex_M23.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M23.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M23.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M23.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M23.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M23.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M23.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M23.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M23.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M23.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M23.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M23.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M23.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M23.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M23.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M23.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M23.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M23.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M23.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M23.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M23.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M23.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M23.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M23.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M23.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M23.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M23.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M23.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M23.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M23.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M23.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M23.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M23.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M23.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M23.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M23.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M23.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M23.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.
FVP_MPS2_Cortex_M23.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M23.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M23.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M23.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M23.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M23.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M23.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M23.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M23.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.

Table 6-3 FVP_MPS2_Cortex-M23 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M23.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M23.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M23.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.4 FVP_MPS2_Cortex-M3

FVP_MPS2_Cortex-M3 contains the following instances:

Table 6-4 FVP_MPS2_Cortex-M3 instances

Name	Type	Description
FVP_MPS2_Cortex_M3	FVP_MPS2_Cortex_M3	-
FVP_MPS2_Cortex_M3.armcortexm3ct	ARM_Cortex-M3	ARM CORTEXM3 CT model.
FVP_MPS2_Cortex_M3.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M3.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M3.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M3.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M3.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M3.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M3.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M3.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M3.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M3.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M3.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M3.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M3.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M3.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M3.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M3.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M3.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M3.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M3.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M3.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M3.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M3.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M3.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M3.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M3.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M3.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M3.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M3.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M3.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M3.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M3.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M3.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M3.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M3.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M3.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M3.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M3.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M3.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M3.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M3.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M3.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M3.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M3.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M3.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-4 FVP_MPS2_Cortex-M3 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M3.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M3.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M3.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M3.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M3.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M3.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.5 FVP_MPS2_Cortex-M33

FVP_MPS2_Cortex-M33 contains the following instances:

Table 6-5 FVP_MPS2_Cortex-M33 instances

Name	Type	Description
FVP_MPS2_Cortex_M33	FVP_MPS2_Cortex_M33	-
FVP_MPS2_Cortex_M33.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.cpu0	ARM_Cortex-M33	ARM CORTEXM33 CT model.
FVP_MPS2_Cortex_M33.cpu1	ARM_Cortex-M33	ARM CORTEXM33 CT model.
FVP_MPS2_Cortex_M33.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M33.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M33.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M33.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M33.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M33.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M33.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M33.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M33.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M33.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M33.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M33.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M33.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M33.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M33.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M33.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M33.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M33.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M33.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M33.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M33.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M33.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M33.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M33.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M33.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M33.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M33.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M33.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M33.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M33.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M33.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M33.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M33.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M33.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M33.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M33.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M33.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M33.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.
FVP_MPS2_Cortex_M33.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M33.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M33.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M33.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M33.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M33.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M33.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M33.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M33.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.

Table 6-5 FVP_MPS2_Cortex-M33 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M33.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M33.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M33.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.6 FVP_MPS2_Cortex-M35P

FVP_MPS2_Cortex-M35P contains the following instances:

Table 6-6 FVP_MPS2_Cortex-M35P instances

Name	Type	Description
FVP_MPS2_Cortex_M35P	FVP_MPS2_Cortex_M35P	-
FVP_MPS2_Cortex_M35P.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.cpu0	ARM_Cortex-M35P	ARM CORTEXM35P CT model.
FVP_MPS2_Cortex_M35P.cpu1	ARM_Cortex-M35P	ARM CORTEXM35P CT model.
FVP_MPS2_Cortex_M35P.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M35P.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M35P.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M35P.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M35P.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M35P.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M35P.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M35P.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M35P.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M35P.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M35P.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M35P.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M35P.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M35P.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M35P.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M35P.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M35P.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M35P.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M35P.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M35P.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M35P.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M35P.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.
FVP_MPS2_Cortex_M35P.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M35P.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M35P.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M35P.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M35P.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M35P.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-6 FVP_MPS2_Cortex-M35P instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M35P.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M35P.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M35P.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M35P.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M35P.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M35P.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.7 FVP_MPS2_Cortex-M4

FVP_MPS2_Cortex-M4 contains the following instances:

Table 6-7 FVP_MPS2_Cortex-M4 instances

Name	Type	Description
FVP_MPS2_Cortex_M4	FVP_MPS2_Cortex_M4	-
FVP_MPS2_Cortex_M4.armcortexm4ct	ARM_Cortex-M4	ARM CORTEXM4 CT model.
FVP_MPS2_Cortex_M4.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M4.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M4.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M4.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M4.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M4.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M4.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M4.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M4.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M4.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M4.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M4.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M4.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M4.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M4.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M4.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M4.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M4.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M4.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M4.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M4.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M4.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M4.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M4.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M4.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M4.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M4.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M4.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M4.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M4.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M4.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M4.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M4.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M4.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M4.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M4.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M4.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M4.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M4.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M4.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M4.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M4.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M4.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M4.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-7 FVP_MPS2_Cortex-M4 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M4.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M4.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M4.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M4.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M4.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M4.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.8 FVP_MPS2_Cortex-M55

FVP_MPS2_Cortex-M55 contains the following instances:

Table 6-8 FVP_MPS2_Cortex-M55 instances

Name	Type	Description
FVP_MPS2_Cortex_M55	FVP_MPS2_Cortex_M55	-
FVP_MPS2_Cortex_M55.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.cpu0	ARM_Cortex-M55	ARM Cortex-M55 CT model.
FVP_MPS2_Cortex_M55.cpu1	ARM_Cortex-M55	ARM Cortex-M55 CT model.
FVP_MPS2_Cortex_M55.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M55.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M55.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M55.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M55.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M55.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M55.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M55.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M55.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M55.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M55.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M55.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M55.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M55.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M55.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M55.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M55.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M55.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M55.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M55.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M55.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M55.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M55.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M55.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M55.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M55.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M55.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M55.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M55.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M55.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M55.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M55.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M55.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M55.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M55.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M55.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M55.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M55.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.
FVP_MPS2_Cortex_M55.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M55.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M55.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M55.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M55.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M55.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M55.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M55.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M55.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.

Table 6-8 FVP_MPS2_Cortex-M55 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M55.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M55.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M55.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

6.9 FVP_MPS2_Cortex-M7

FVP_MPS2_Cortex-M7 contains the following instances:

Table 6-9 FVP_MPS2_Cortex-M7 instances

Name	Type	Description
FVP_MPS2_Cortex_M7	FVP_MPS2_Cortex_M7	-
FVP_MPS2_Cortex_M7.armcortexm7ct	ARM_Cortex-M7	ARM CORTEXM7 CT model.
FVP_MPS2_Cortex_M7.clk25Mhz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.clk25khz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.default_ahb_slave	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M7.fvp_mps2	FVP_MPS2	MPS2 DUT.
FVP_MPS2_Cortex_M7.fvp_mps2.GPIO0	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M7.fvp_mps2.GPIO1	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M7.fvp_mps2.GPIO2	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M7.fvp_mps2.GPIO3	CMSDK_GPIO	ARM PrimeCell General Purpose Input/Output.
FVP_MPS2_Cortex_M7.fvp_mps2.GPIO_connection_test	GPIO_Connection_Test	-
FVP_MPS2_Cortex_M7.fvp_mps2.GPIO_connection_test.GPIO0_port_trans	GPIO_Port_Transfer	-
FVP_MPS2_Cortex_M7.fvp_mps2.GPIO_connection_test.GPIO1_port_test	GPIO1_Connection_Test	-
FVP_MPS2_Cortex_M7.fvp_mps2.PSRAM	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.PSRAM_M7	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.UART0	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M7.fvp_mps2.UART0.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.UART1	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M7.fvp_mps2.UART1.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.UART2	CMSDK_UART	ARM CMSDK UART Module.
FVP_MPS2_Cortex_M7.fvp_mps2.UART2.clk_divider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.VGA_interface	MPS2_VGA	VGA display interface between main bus and visualisation.
FVP_MPS2_Cortex_M7.fvp_mps2.ahb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.ahb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.apb_ppc_iotss_expansion0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.apb_ppc_iotss_expansion1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.apb_ppc_iotss_expansion2	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.clock50Hz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.cmsdk_sysctrl	CMSDK_SysCtrl	Cortex-M Simple System Control.
FVP_MPS2_Cortex_M7.fvp_mps2.cmsdk_watchdog	CMSDK_Watchdog	ARM Watchdog Module.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.cpu_wait_or_gate_0	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.cpu_wait_or_gate_1	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.dbgen_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.dma0	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M7.fvp_mps2.dma0.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M7.fvp_mps2.dma0.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M7.fvp_mps2.dma0.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M7.fvp_mps2.dma0.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.dma0_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M7.fvp_mps2.dma0_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M7.fvp_mps2.dma1	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M7.fvp_mps2.dma1.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.dma1.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M7.fvp_mps2.dma1.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M7.fvp_mps2.dma1.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.dma1_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M7.fvp_mps2.dma1_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M7.fvp_mps2.dma2	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).
FVP_MPS2_Cortex_M7.fvp_mps2.dma2.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M7.fvp_mps2.dma2.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M7.fvp_mps2.dma2.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M7.fvp_mps2.dma2.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.dma2_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M7.fvp_mps2.dma2_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M7.fvp_mps2.dma3	*PL080_DMAC*	ARM PrimeCell DMA Controller(PL080/081).

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.dma3.timer	*ClockTimerThread*	A ClockTimerThread(64) is a drop-in replacement for a ClockTimer(64) component. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M7.fvp_mps2.dma3.timer.timer	*ClockTimerThread64*	A ClockTimerThread64 is a drop-in replacement for ClockTimer64. The main difference to the ClockTimer component is that the ClockTimerThread runs the signal() callback from a proper scheduler thread. This mean that the signal() function may directly or indirectly invoke wait() functions to wait for time or events. This is not allowed for the ClockTimer component which does not use a thread. Components which issue bus transactions from within the timer signal() callback must use ClockTimerThread(64) rather than ClockTimer(64).
FVP_MPS2_Cortex_M7.fvp_mps2.dma3.timer.timer.thread	*SchedulerThread*	A SchedulerThread instance represents a co-routine thread in the simulation.
FVP_MPS2_Cortex_M7.fvp_mps2.dma3.timer.timer.thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.dma3_idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M7.fvp_mps2.dma3_securitymodifier	SecurityModifier	-
FVP_MPS2_Cortex_M7.fvp_mps2.exclusive_monitor_psram	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.exclusive_monitor_psram_iotss	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.exclusive_monitor_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.exclusive_monitor_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M7.fvp_mps2.expansion_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M7.fvp_mps2.fpga_sysctrl	FPGA_SysCtrl	FPGA SysCtrl timers LEDs and switches.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.fpga_sysctrl.callBack100HzCounter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.fpga_sysctrl.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.gpio_0_or_2	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.gpio_1_or_3	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.hostbridge	*HostBridge*	Host Socket Interface Component.
FVP_MPS2_Cortex_M7.fvp_mps2.mem_switch_extra_psram_iotss	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.mem_switch_extra_psram_mps2	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.mpc_iotss_ssram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.mpc_iotss_ssram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.mpc_iotss_ssram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.mpc_iotss_ssram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_audio	MPS2_Audio	MPS2 Audio.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_exclusive_monitor_zbtsram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_exclusive_monitor_zbtsram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_lcd	MPS2_LCD	MPS2 LCD I2C interface.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_mpc_iotss_ssram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_mpc_iotss_ssram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.mps2_visualisation	MPS2_Visualisation	Display window for MPS2 using Visualisation library.
FVP_MPS2_Cortex_M7.fvp_mps2.niden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.pl022_ssp_mps2	PL022_SSP_MPS2	ARM PrimeCell Synchronous Serial Port(PL022).
FVP_MPS2_Cortex_M7.fvp_mps2.pl022_ssp_mps2.prescaler	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.platform_bus_switch	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.platform_switch_dma0	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.platform_switch_dma1	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M7.fvp_mps2.smsc_91c111	*SMSC_91C111*	SMSC 91C111 ethernet controller.
FVP_MPS2_Cortex_M7.fvp_mps2.spiden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.spniden_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200	SSE200	SSE-200 subsystem.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.acg_cpu0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.acg_cpu1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.acg_sram0	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.acg_sram1	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.acg_sram2	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.acg_sram3	IoTSS_AccessControlGate	IoT Subsystem Access Control Gate.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.apb_ppc_iotss_subsystem0	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.apb_ppc_iotss_subsystem1	IoTSS_PeripheralProtectionController	IoT Subsystem Peripheral Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.bus_error_warning_memory	*WarningMemory*	Memory that prints warnings, and RAZ/WIs or aborts.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.clock32kHz	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.clockdivider	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cmsdk_dualtimer	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cmsdk_dualtimer.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cmsdk_dualtimer.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cmsdk_dualtimer.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cmsdk_dualtimer.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cordio_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cpu0core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cpu0dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cpu1core_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.cpu1dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.crypto_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.dbg_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram0	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram1	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram2	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.exclusive_monitor_iotss_internal_sram3	*PVBusExclusiveMonitor*	Global exclusive monitor.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.exclusive_squasher	*PVBusExclusiveSquasher*	Squashes the exclusive attribute on bus transactions.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.idau_labeller	LabellerIdauSecurity	-
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.iotss_cpuidentity	IoTSS_CPUIdentity	IoT Subsystem CPU_IDENTITY registers.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.iotss_internal_sram0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.iotss_internal_sram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.iotss_internal_sram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.iotss_internal_sram3	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.iotss_systemcontrol	IoTSS_SystemControl	IoT Subsystem System Control registers.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.iotss_systeminfo	IoTSS_SystemInfo	IoT Subsystem System Information registers.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mem_switch_internal_sram	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mem_switch_internal_sram_mpc	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mem_switch_ppu	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mhu0	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mhu1	IoTSS_MessageHandlingUnit	IoT Subsystem Message Handling Unit.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram0	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram0.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram1	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram1.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram2	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram2.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram3	IoTSS_MemoryProtectionController	IoT Subsystem Memory Protection Controller.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.mpc_iotss_internal_sram3.gating_disabled_thread_event	*SchedulerThreadEvent*	A SchedulerThreadEvent instance is an auto-reset boolean condition variable other threads can wait on.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.nmi_or_gate	*OrGate*	Or Gate.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.nonsecure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.ram0_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.ram1_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.ram2_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.ram3_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.s32k_timer	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.s32k_timer.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.s32k_timer.counter	*CounterModule*	Internal component used by SP804 Timer module.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.s32k_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.secure_control_register_block	MPS2_SecureCtrl	MPS2 Secure Control Register Block.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.secure_watchdog	CMSDK_Watchdog	ARM Watchdog Module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.signal_router	SignalRouter	Signal router.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.sys_ppu	*PPUv0*	ARM Power Policy Unit (PPU) architectural model.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.timer0	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.timer0.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.timer0.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.timer1	CMSDK_Timer	ARM Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.timer1.clk_div	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.sse200.timer1.counter	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.ssram1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.ssram2	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.stub0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.stub1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.stub_i2c1	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.stub_i2s	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.stub_spi0	*RAMDevice*	RAM device, can be dynamic or static ram.
FVP_MPS2_Cortex_M7.fvp_mps2.stub_spi2	*RAMDevice*	RAM device, can be dynamic or static ram.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer	SVOS_DualTimer	-
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer0	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer0.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer0.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer1	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer1.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer1.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer2	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.

Table 6-9 FVP_MPS2_Cortex-M7 instances (continued)

Name	Type	Description
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer2.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer2.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer3	CMSDK_DualTimer	ARM Dual-Timer Module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div0	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer3.clk_div1	*ClockDivider*	A ClockDivider is a library component that takes a ClockSignal on its input port (which could come from the output of a MasterClock, or from another ClockDivider), and generates a new ClockSignal on its output port, representing a clock frequency that is related to the input clock by the ratio of the multiply and divide parameters.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter0	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.svos_dualtimer.svos_dualtimer3.counter1	*CounterModule*	Internal component used by SP804 Timer module.
FVP_MPS2_Cortex_M7.fvp_mps2.switch_svos_dualtimer	*PVBusRouter*	Allow transactions to be routed arbitrarily.
FVP_MPS2_Cortex_M7.fvp_mps2.telnetterminal0	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M7.fvp_mps2.telnetterminal1	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M7.fvp_mps2.telnetterminal2	*TelnetTerminal*	Telnet terminal interface.
FVP_MPS2_Cortex_M7.fvp_mps2.touchscreen_interface	MPS2_TouchScreen	MPS2 Touch Screen.
FVP_MPS2_Cortex_M7.fvp_mps2.uart_overflows_or_gate	*OrGate*	Or Gate.

Chapter 7
Arm® Neoverse™ reference design FVPs

This chapter describes the Arm Neoverse™ N1 edge and Arm Neoverse E1 edge reference design FVPs. These FVPs are collectively referred to as RD-N1-E1-edge FVPs.

A reference design is a collection of resources, including documentation, a software stack, and FVPs, that describe and model the design choices and performance for recommended configurations of a typical Arm-based subsystem.

The RD-N1-E1-edge FVPs drive system architecture and software standardization. They provide software and binaries of proprietary firmware that reduce the amount of work that is required for SoC development.

Note

Arm is working to make more content available for these FVPs. To find out more about reference designs, or to contact Arm about them, see *Reference Design*.

It contains the following sections:

- *7.1 About the FVPs* on page 7-1125.
- *7.2 Block diagrams for RD-N1-E1-edge* on page 7-1126.
- *7.3 FVP peripherals* on page 7-1128.

7.1 About the FVPs

The RD-N1-E1-edge FVPs model many of the Arm IP components in the RD-N1-E1-edge design.

The RD-N1-E1-edge FVPs model the following RD-N1-E1-edge configurations:

Config1
N1 2xMP4, 512KB L2 cache per core, 4x2 mesh, 2xDMC.

Config2
E1 2xMP8, 256KB L2 cache per core, 4x2 mesh, 2xDMC.

Config3
Dual-chip. Two Config1 subsystems linked by CMN-600 CML.

The diagrams in *7.2 Block diagrams for RD-N1-E1-edge* on page 7-1126 show the IP components that RD-N1-E1-edge describes. Not all of these components are modeled by these FVPs. The following components are modeled:

- N1 2xMP4 or E1 2xMP8 cores.
- System Control Processor (SCP).
- Management Control Processor (MCP).
- CMN-600.
- Multiple NIC-450 interconnects, although NIC-450 is replaced with a simple bus model.
- Memory access path towards DRAM, including DMC-620 memory controllers.

The following components are not modeled:

- CoreSight™ SoC.
- ELA-500.
- USB.

7.2 Block diagrams for RD-N1-E1-edge

The following diagrams show the composition of RD-N1-E1-edge systems.

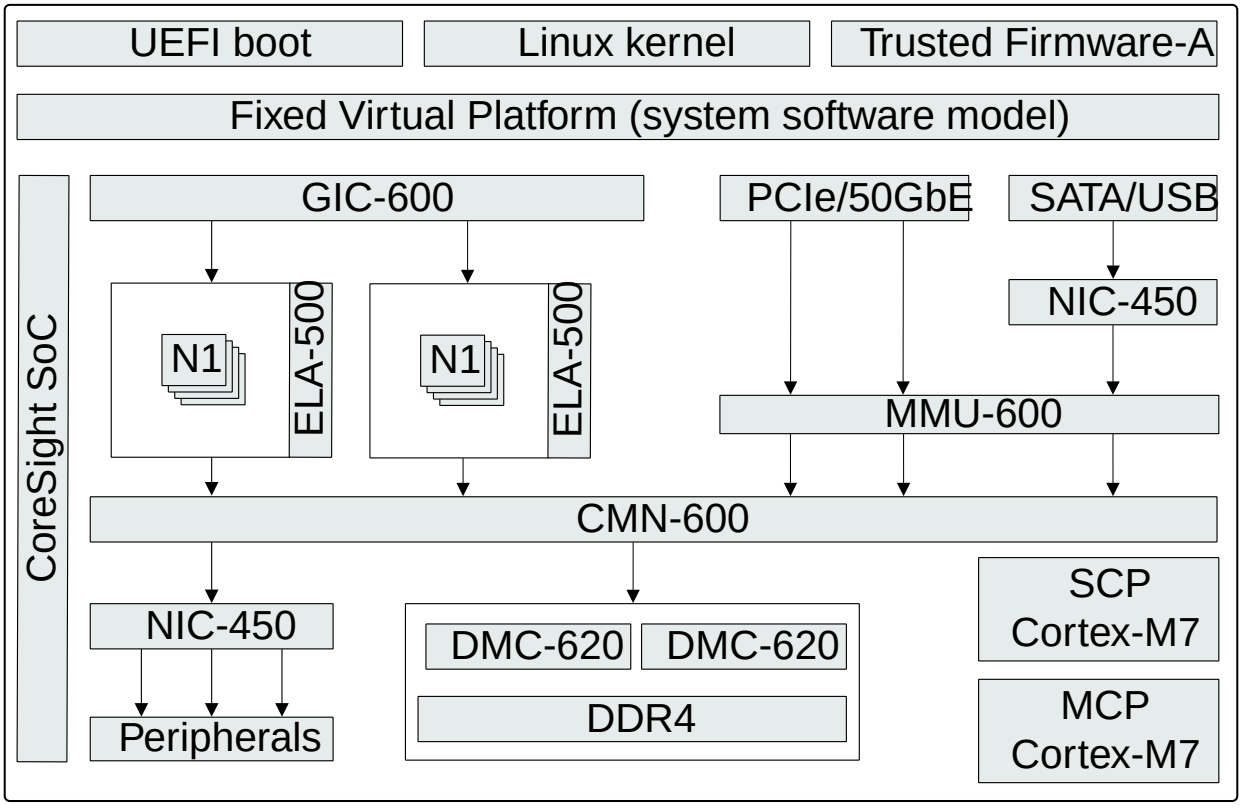

Figure 7-1 Block diagram for Neoverse N1 edge reference design

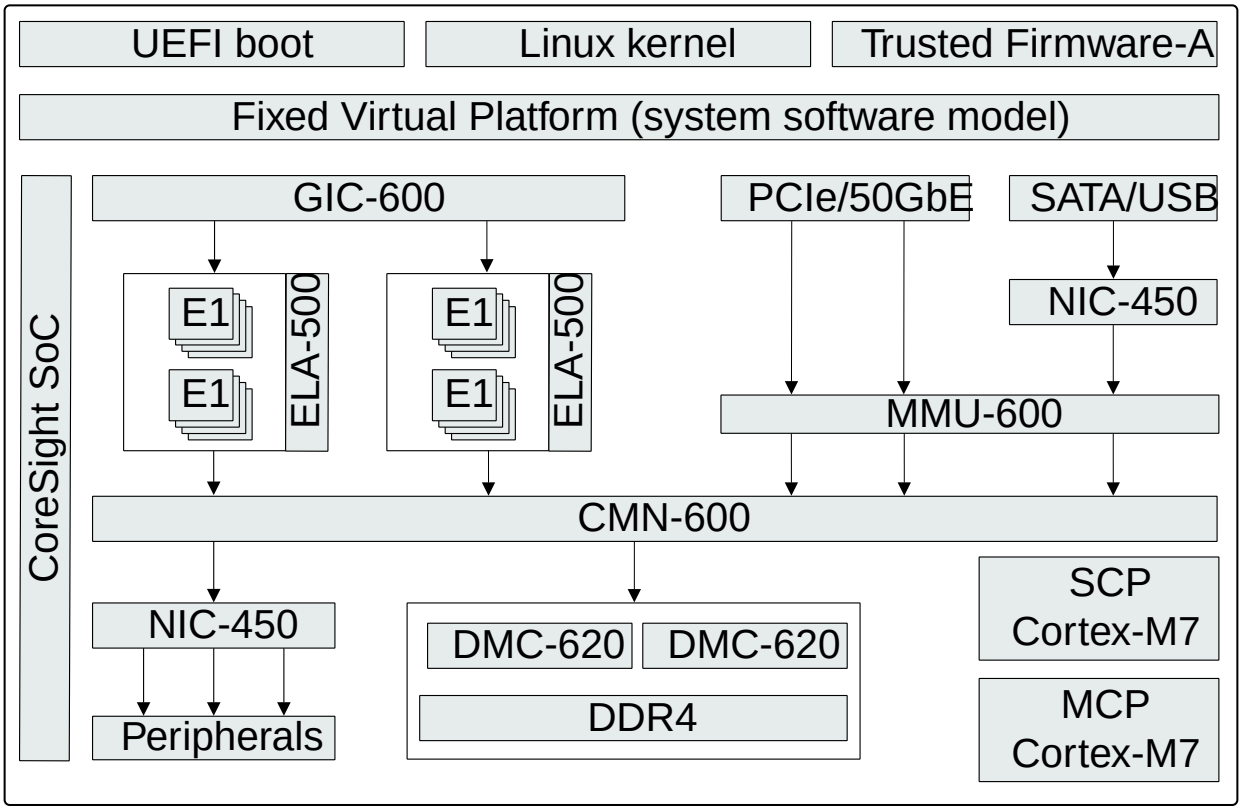

Figure 7-2 Block diagram for Neoverse E1 edge reference design

7.3 FVP peripherals

The RD-N1-E1-edge FVPs include peripherals that the software payload requires to run.

These peripherals are organized in two layers:

SoC

The SoC peripherals represent peripherals that may be added to a compute subsystem in a SoC design.

Board

The board peripherals represent peripherals that may be present on the board onto which the SoC is mounted.

The RD-N1-E1-edge SoC model and board model are based on the Juno Arm Development Platform (ADP).

In Config3, each compute subsystem has its own SoC and board layers. In this two-chip configuration, each chip is assigned the following memory regions:

Chip 0

`0x000_0000_0000-0x3FF_FFFF_FFFF`

Chip 1

`0x400_0000_0000-0x7FF_FFFF_FFFF`

A sideband communication channel is required to coordinate multi-chiplet software boot over CMN-600. The FVP implements this using the MHU device, but Arm recommends using a solution such as I2C in hardware.

This section contains the following subsections:

- *7.3.1 SoC peripherals memory map* on page 7-1128.
- *7.3.2 Board peripherals memory map* on page 7-1129.
- *7.3.3 Interrupt maps* on page 7-1130.

7.3.1 SoC peripherals memory map

This table shows the memory map for the SoC peripherals in the RD-N1-E1-edge FVPs.

Note

The SoC peripherals area in the RD_N1_E1_edge memory map is mapped to an Expansion AXI region. Therefore, these mappings are not defined in the reference design.

Table 7-1 SoC peripherals memory map

Name	Base address	Size	Description
SMC interface	`0x00_0800_0000`	368MB	Routed to board
SMMUv3	`0x00_2B40_0000`	1MB	-
PCIe config	`0x00_6000_0000`	16MB	-
PCIe memory	`0x00_7000_0000`	132MB	-
DMA MMU-400	`0x00_7FB0_0000`	64KB	-
HDLCD1 MMU-400	`0x00_7FB1_0000`	64KB	-
HDLCD0 MMU-400	`0x00_7FB2_0000`	64KB	-
DDR3 PHY 0	`0x00_7FB6_0000`	64KB	Dummy APB

Table 7-1 SoC peripherals memory map (continued)

Name	Base address	Size	Description
DDR3 PHY 1	0x00_7FB7_0000	64KB	Dummy APB
DDR3 PHY 2	0x00_7FB8_0000	64KB	Dummy APB
DDR3 PHY 3	0x00_7FB9_0000	64KB	Dummy APB
SoC interconnect NIC-400 GPV	0x00_7FD0_0000	1MB	-
Surge detector	0x00_7FE5_0000	4KB	Dummy APB
TRNG	0x00_7FE6_0000	4KB	-
Trusted non-volatile counters	0x00_7FE7_0000	4KB	-
Trusted Root-Key storage	0x00_7FE8_0000	4KB	-
Secure I2C	0x00_7FE9_0000	256B	A Secure I2C component does not exist in the FVP. Instead, a PL061_GPIO is mapped as a dummy component. It is not functional as a GPIO.
DMA non-secure	0x00_7FF0_0000	4KB	-
DMA secure	0x00_7FF1_0000	4KB	-
HDLCD1	0x00_7FF5_0000	4KB	-
HDLCD0	0x00_7FF6_0000	4KB	-
UART 1	0x00_7FF7_0000	4KB	-
UART 0	0x00_7FF8_0000	4KB	-
I2S	0x00_7FF9_0000	1KB	An I2S component does not exist in the FVP. Instead, a PL061_GPIO is mapped as a dummy component. It is not functional as a GPIO.
I2C	0x00_7FFA_0000	0x256B	An I2C component does not exist in the FVP. Instead, a PL061_GPIO is mapped as a dummy component. It is not functional as a GPIO.
PL352	0x00_7FFD_0000	4KB	PL354
AP configuration	0x00_7FFE_0000	4KB	GPR
System override registers	0x00_7FFF_0000	4KB	-
PCIe memory	0x05_0000_0000	12GB	-

Note

The following devices are discoverable on the PCIe bus:

- Virtio block device x 2.
- AHCI controller with attached SATA disk.

7.3.2 Board peripherals memory map

This table shows the memory map for the board peripherals in the RD-N1-E1-edge FVPs.

Note

The board peripherals area in the RD_N1_E1_edge memory map is mapped to an Expansion AXI region. Therefore, these mappings are not defined in the reference design.

Table 7-2 Board peripherals memory map

Name	Base address	Size	Description
NOR Flash 0	0x00_0800_0000	64MB	-
NOR Flash 1	0x00_0C00_0000	64MB	-
NOR Flash 2	0x00_1000_0000	64MB	-
Ethernet	0x00_1800_0000	64MB	SMSC 91C111
System registers	0x00_1C01_0000	64KB	-
MCI	0x00_1C05_0000	64KB	PL180
KMI 0	0x00_1C06_0000	64KB	PL050
KMI 1	0x00_1C07_0000	64KB	PL050
UART 0	0x00_1C09_0000	64KB	PL011
UART 1	0x00_1C0A_0000	64KB	PL011
Watchdog	0x00_1C0F_0000	64KB	SP805
Dual timer	0x00_1C11_0000	64KB	SP804
Virtio block device	0x00_1C13_0000	64KB	-
Virtio net device	0x00_1C15_0000	64KB	-
RTC	0x00_1C17_0000	64KB	PL031
GPIO 0	0x00_1C1D_0000	64KB	PL061_GPIO is mapped as a dummy component in the FVP. It is not functional as a GPIO.
GPIO 1	0x00_1C1E_0000	64KB	PL061_GPIO is mapped as a dummy component in the FVP. It is not functional as a GPIO.
DRAM	0x00_8000_0000	2GB	-
DRAM	0x80_8000_0000	6GB	-

7.3.3 Interrupt maps

These tables show the interrupt IDs and sources for the FVP peripherals.

Table 7-3 Interrupt map at the SoC layer

Interrupt ID	Source	Description
147	UART 0	-
148	UART 1	-
171	TRNG	-

Table 7-4 Interrupt map at the board layer

Interrupt ID	Source	Description
111	Ethernet	-
132	RTC	-
133	UART 0	-
134	UART 1	-

Table 7-4 Interrupt map at the board layer (continued)

Interrupt ID	Source	Description
140	VFS2	-
202	Virtio	-
204	Virtio net device	-
228	Watchdog	-
229	KMI 0	-
230	Dual timer	Interrupts 0 and 1
231	System registers ethernet IRQ	-