



Arm[®] Morello System Development Platform

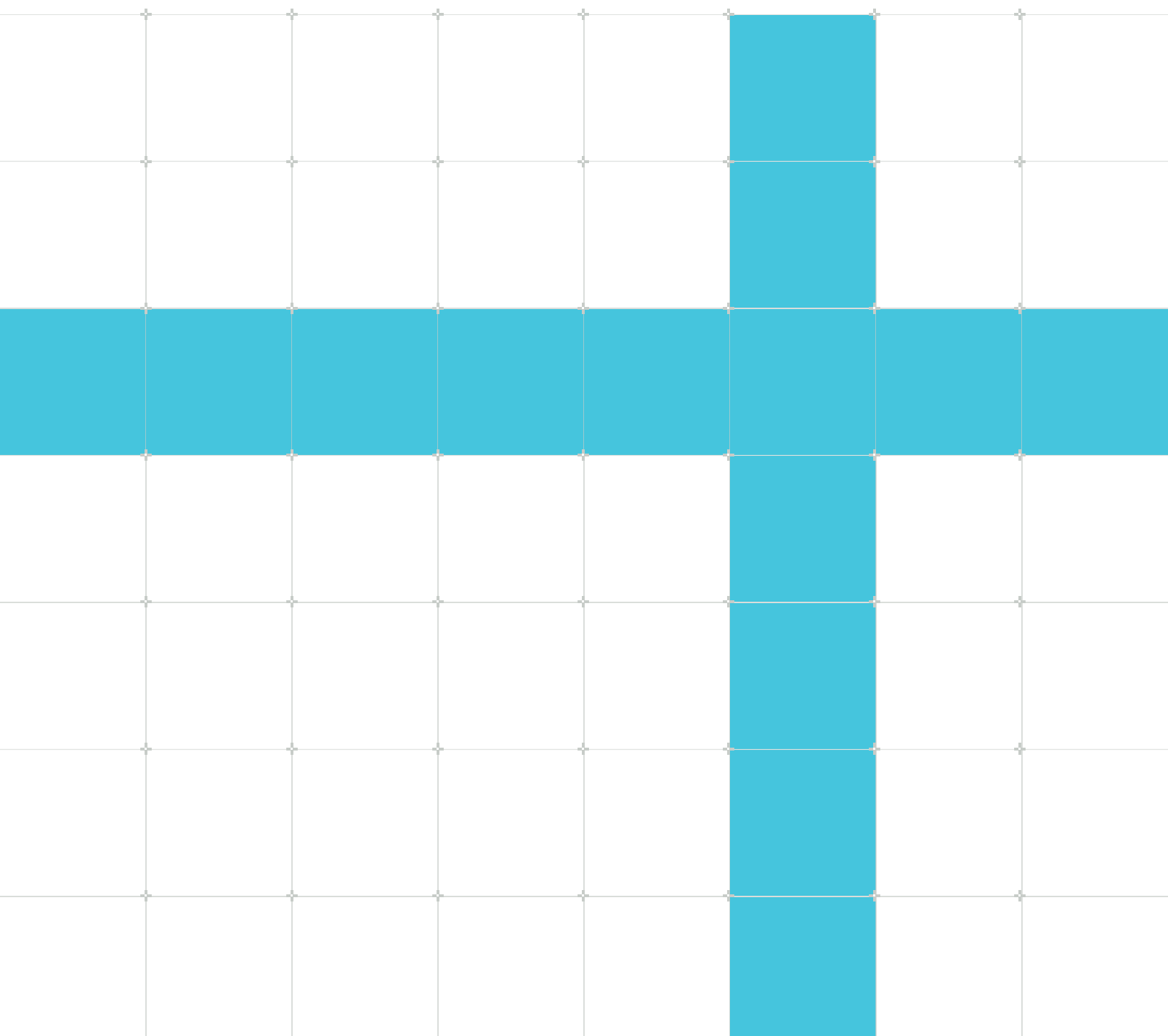
Revision: r0p0

Technical Reference Manual

Non-Confidential

Issue 03

Copyright © 2020–2022 Arm Limited (or its affiliates). 102278_0000_03_en
All rights reserved.



Arm® Morello System Development Platform

Technical Reference Manual

Copyright © 2020–2022 Arm Limited (or its affiliates). All rights reserved.

Release information

Document history

Issue	Date	Confidentiality	Change
0000-01	29 October 2020	Non-Confidential	Beta release
0000-02	19 January 2021	Non-Confidential	Second Beta release
0000-03	18 January 2022	Non-Confidential	EAC release

Proprietary Notice

This document is protected by copyright and other related rights and the practice or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of Arm. No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether implementations infringe any third party patents.

THIS DOCUMENT IS PROVIDED “AS IS”. ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, Arm makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, third party patents, copyrights, trade secrets, or other rights.

This document may include technical inaccuracies or typographical errors.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

If any of the provisions contained in these terms conflict with any of the provisions of any click through or signed written agreement covering this document with Arm, then the click through or signed written agreement prevails over and supersedes the conflicting provisions of these terms. This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

The Arm corporate logo and words marked with ® or ™ are registered trademarks or trademarks of Arm Limited (or its subsidiaries in the US and/or elsewhere). All rights reserved. Other brands and names mentioned in this document may be the trademarks of their respective owners. Please follow Arm’s trademark usage guidelines at <https://www.arm.com/company/policies/trademarks>.

Copyright © 2020–2022 Arm Limited (or its affiliates). All rights reserved.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(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.

Feedback

Arm® welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on <https://support.developer.arm.com>

To provide feedback on the document, fill the following survey: <https://developer.arm.com/documentation-feedback-survey>.

Inclusive language commitment

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

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

To report offensive language in this document, email terms@arm.com.

Conformance Notices

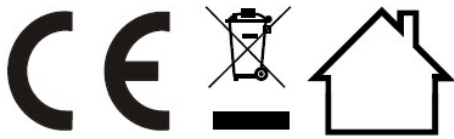
This section contains conformance notices.

Federal Communications Commission Notice

This device is test equipment and so is exempt from part 15 of the FCC Rules under section 15.103 (c).

CE Conformity

The Declaration of Conformity applies to all EU member states.
The Declaration of Conformity is available on request.



The Waste Electrical and Electronic Equipment (WEEE) marking, that is, the crossed out wheeled-bin figure, indicates that this product must not be disposed of with general waste within the European Union. To prevent possible harm to the environment from uncontrolled waste disposal, the user is required to recycle the product responsibly to promote reuse of material resources. To comply with EU law, you must dispose of the product in one of the following ways:

- Return it to the distributor where it was purchased. The distributor is required to arrange free collection when requested.
- Recycle it using local WEEE recycling facilities. These facilities are now very common and might provide free collection.
- If purchased directly from Arm, Arm provides free collection. Please email weee@arm.com for instructions.
- End-of-Life products can be disposed of safely using an Approved Authorized Treatment Facility (AATF). To support safe disposal, Arm has partnered with B2B Compliance. B2B can be contacted at the following weblink: <https://b2bcompliance.org.uk>

During the lifetime of the product, you are advised to:

- Inspect the product regularly to ensure that it is in good working order.
- Ensure that the product is free from dust and debris that might cause damage.
- Clean the product with an air duster when necessary.
- Observe the guidelines described in [2.3.3 Preventing damage](#) on page 21.
- Power down the system when not in use.
- Observe ESD precautions when handling the product.

The product can radiate Radio Frequency Interference (RFI) or Electromagnetic Interference (EMI) and might cause harmful interference to radio communications. There is no guarantee that interference cannot occur in a particular installation. If you suspect that this equipment is causing interference to other equipment, you are encouraged to try to correct the interference by one or more of the following measures:

- Ensure attached cables do not lie across any sensitive equipment.
- Increase the distance between the product and the receiver.
- Connect the equipment to an outlet on a circuit different from that to which the product is connected.
- Consult Arm for help.

The product can be sensitive to Radio Frequency Interference (RFI) or Electromagnetic Interference (EMI) which might cause incorrect operation of the product:

- Avoid using the product near sources of EMI.
- Never use the product in Safety-Critical-Systems (SCS), or Life-Critical-Systems (LCS).

Arm recommends that, wherever possible, shielded interface cables be used.

Contents

1 Introduction.....	14
1.1 Product revision status.....	14
1.2 Intended audience.....	14
1.3 Conventions.....	14
1.4 Additional reading.....	16
2 Overview.....	18
2.1 About the Morello SDP.....	18
2.2 The Morello SDP at a glance.....	19
2.3 Measures to prevent damage and ensure safety.....	22
2.3.1 Ensuring safety.....	22
2.3.2 Operating temperature.....	22
2.3.3 Preventing damage.....	22
2.4 Getting started.....	23
2.5 Accessing the ATX power cables.....	24
3 Hardware description.....	25
3.1 Morello SDP hardware.....	25
3.2 Major components of the Morello SoC.....	29
3.3 External power.....	32
3.3.1 Overview of power scheme.....	33
3.3.2 Power islands.....	33
3.4 Clocks.....	34
3.4.1 Overview of clocks.....	34
3.4.2 SoC clocks.....	35
3.4.3 Clock programming and control.....	38
3.4.4 IOFPGA clocks.....	41
3.5 Resets.....	41
3.6 IOFPGA.....	43
3.6.1 Overview of IOFPGA.....	43
3.6.2 IOFPGA interrupts.....	46
3.7 PCI Express and CCIX systems.....	47
3.7.1 Overview of PCIe and CCIX systems.....	48

3.7.2 PCI Express and CCIX expansion slots.....	49
3.7.3 SATA 3.0 ports.....	49
3.7.4 Gigabit Ethernet port.....	49
3.7.5 USB 3.0 ports.....	49
3.8 UARTs.....	50
3.9 LEDs, switches, and buttons.....	53
3.9.1 MCC system LEDs.....	53
3.9.2 PCC system LEDs.....	54
3.9.3 Miscellaneous LEDs.....	54
3.9.4 Push buttons and switches.....	55
3.10 Debug.....	56
3.11 Embedded ULINKplus.....	56
3.11.1 ULINK mux control.....	56
4 Configuration.....	58
4.1 Overview of the configuration process.....	58
4.2 Powerup and powerdown sequences.....	59
4.3 Configuration files.....	60
4.3.1 Overview of configuration files and microSD card directory structure.....	61
4.3.2 config.txt board configuration file.....	62
4.3.3 Contents of the MB directory.....	63
4.3.4 Contents of the SOFTWARE subdirectory.....	65
4.4 Configuration switches.....	66
4.4.1 Use of configuration switches.....	66
4.4.2 Remote UART configuration.....	66
4.5 Use of reset push buttons.....	67
4.6 Command-line interface.....	68
4.6.1 Overview of the Morello SDP MCC command-line interface.....	68
4.6.2 MCC main command menu.....	68
4.6.3 MCC debug menu.....	69
4.6.4 EEPROM menu.....	70
5 Programmers model.....	72
5.1 About this programmers model.....	72
5.2 Morello SDP memory maps.....	72
5.2.1 Application Processor memory map.....	73
5.2.2 Application Processor subsystem peripherals memory map.....	74

5.2.3 Manageability Control Processor memory map.....	78
5.2.4 Manageability Control Processor peripherals memory map.....	79
5.2.5 System Control Processor memory map.....	82
5.2.6 System Control Processor peripherals memory map.....	83
5.2.7 CoreSight system memory map.....	85
5.2.8 IOFPGA memory map.....	88
5.3 Morello SoC interrupt maps.....	89
5.3.1 Application Processor interrupt map.....	90
5.3.2 System Control Processor interrupt map.....	93
5.3.3 Manageability Control Processor interrupt map.....	97
5.4 System Security Control registers.....	99
5.4.1 System Security Control registers summary.....	99
5.4.2 SSC_DBGCFG_STAT Register.....	100
5.4.3 SSC_DBGCFG_SET Register.....	101
5.4.4 SSC_DBGCFG_CLR Register.....	102
5.4.5 SSC_AUXDBGCFG Register.....	104
5.4.6 SSC_GPRETN Register.....	105
5.4.7 SSC_VERSION Register.....	105
5.4.8 SSC_SW_SCRATCH Registers.....	106
5.4.9 SSC_SW_CAP Registers.....	106
5.4.10 SSC_SW_CAPCTRL Register.....	107
5.4.11 SSC_CHIPID_ST Register.....	107
5.4.12 SSC_PID4 Register.....	108
5.4.13 SSC_PID0 Register.....	109
5.4.14 SSC_PID1 Register.....	109
5.4.15 SSC_PID2 Register.....	110
5.4.16 SSC_COMPID0 Register.....	110
5.4.17 SSC_COMPID1 Register.....	111
5.4.18 SSC_COMPID2 Register.....	111
5.4.19 SSC_COMPID3 Register.....	112
5.5 Serial Configuration Control registers.....	112
5.5.1 Serial Configuration Control registers summary.....	113
5.5.2 PMCLK_DIV Register.....	115
5.5.3 SYSAPBCLK_CTRL Register.....	116
5.5.4 SYSAPBCLK_DIV Register.....	117
5.5.5 IOFPGA_TMIF2XCLK_CTRL Register.....	118

5.5.6 IOFPGA_TMIF2XCLK_DIV Register.....	119
5.5.7 IOFPGA_TSIF2XCLK_CTRL Register.....	120
5.5.8 IOFPGA_TSIF2XCLK_DIV Register.....	120
5.5.9 SCPNICCLK_CTRL Register.....	121
5.5.10 SCPNICCLK_DIV Register.....	122
5.5.11 SCPI2CCLK_CTRL Register.....	123
5.5.12 SCPI2CCLK_DIV Register.....	124
5.5.13 SCPQSPICLK_CTRL Register.....	125
5.5.14 SCPQSPICLK_DIV Register.....	125
5.5.15 SENSORCLK_CTRL Register.....	126
5.5.16 SENSORCLK_DIV Register.....	127
5.5.17 MCPNICCLK_CTRL Register.....	128
5.5.18 MCPNICCLK_DIV Register.....	129
5.5.19 MCPI2CCLK_CTRL Register.....	130
5.5.20 MCPI2CCLK_DIV Register.....	130
5.5.21 MCPQSPICLK_CTRL Register.....	131
5.5.22 MCPQSPICLK_DIV Register.....	132
5.5.23 PCIEAXICLK_CTRL Register.....	133
5.5.24 PCIEAXICLK_DIV Register.....	134
5.5.25 CCIXAXICLK_CTRL Register.....	135
5.5.26 CCIXAXICLK_DIV Register.....	135
5.5.27 PCIEAPBCLK_CTRL Register.....	136
5.5.28 PCIEAPBCLK_DIV Register.....	137
5.5.29 CCIXAPBCLK_CTRL Register.....	138
5.5.30 CCIXAPBCLK_DIV Register.....	139
5.5.31 SYS_CLK_EN Register.....	140
5.5.32 CPU0_PLL_CTRL0 Register.....	142
5.5.33 CPU0_PLL_CTRL1 Register.....	143
5.5.34 CPU1_PLL_CTRL0 Register.....	144
5.5.35 CPU1_PLL_CTRL1 Register.....	145
5.5.36 CLUS_PLL_CTRL0 Register.....	146
5.5.37 CLUS_PLL_CTRL1 Register.....	147
5.5.38 SYS_PLL_CTRL0 Register.....	148
5.5.39 SYS_PLL_CTRL1 Register.....	149
5.5.40 DMC_PLL_CTRL0 Register.....	150
5.5.41 DMC_PLL_CTRL1 Register.....	151

5.5.42 INT_PLL_CTRL0 Register.....	152
5.5.43 INT_PLL_CTRL1 Register.....	153
5.5.44 GPU_PLL_CTRL1 Register.....	154
5.5.45 GPU_PLL_CTRL0 Register.....	155
5.5.46 DPU_PLL_CTRL1 Register.....	156
5.5.47 DPU_PLL_CTRL0 Register.....	157
5.5.48 PXL_PLL_CTRL1 Register.....	158
5.5.49 PXL_PLL_CTRL0 Register.....	159
5.5.50 SYS_MAN_RESET Register.....	160
5.5.51 BOOT_CTL Register.....	162
5.5.52 BOOT_CTRL_STA Register.....	163
5.5.53 SCP_BOOT_ADR Register.....	163
5.5.54 MCP_BOOT_ADR Register.....	164
5.5.55 PLATFORM_CTRL Register.....	164
5.5.56 TARGETIDAPP Register.....	165
5.5.57 TARGETIDSCP Register.....	166
5.5.58 TARGETIDMCP Register.....	166
5.5.59 BOOT_GPR0 Register.....	167
5.5.60 BOOT_GPR1 Register.....	167
5.5.61 BOOT_GPR2 Register.....	168
5.5.62 BOOT_GPR3 Register.....	168
5.5.63 BOOT_GPR4 Register.....	169
5.5.64 BOOT_GPR5 Register.....	170
5.5.65 BOOT_GPR6 Register.....	170
5.5.66 BOOT_GPR7 Register.....	171
5.5.67 INSTANCE_ID Register.....	171
5.5.68 PCIE_BOOT_CTRL Register.....	172
5.5.69 GPU_CTRL Register.....	173
5.5.70 DBG_AUTHN_CTRL Register.....	173
5.5.71 CTIO_CTRL Register.....	174
5.5.72 CTI1_CTRL Register.....	175
5.5.73 CTI0TO3_CTRL Register.....	175
5.5.74 MCP_WDOGCTI_CTRL Register.....	176
5.5.75 SCP_WDOGCTI_CTRL Register.....	176
5.5.76 DBGEXPCTI_CTRL Register.....	177
5.5.77 PCIE_PM_CTRL Register.....	178

5.5.78 CCIX_PM_CTRL Register.....	178
5.5.79 SCDBG_CTRL Register.....	179
5.5.80 EXP_IF_CTRL Register.....	181
5.5.81 RO_CTRL Register.....	182
5.5.82 CMN_CCIX_CTRL Register.....	183
5.5.83 STM_CTRL Register.....	184
5.5.84 AXI_OVRD_PCIE Register.....	185
5.5.85 AXI_OVRD_CCIX Register.....	186
5.5.86 AXI_OVRD_TSIF Register.....	187
5.5.87 GPU_TEXFMTENABLE Register.....	187
5.5.88 TRACE_PAD_CTRL0 Register.....	188
5.5.89 TRACE_PAD_CTRL1 Register.....	190
5.5.90 IOFPGA_TMIF_PAD_CTRL Register.....	191
5.5.91 IOFPGA_TSIF_PAD_CTRL Register.....	193
5.5.92 DISPLAY_PAD_CTRL0 Register.....	195
5.5.93 DISPLAY_PAD_CTRL1 Register.....	197
5.5.94 APB_CTRL_CLR Register.....	199
5.5.95 PID4 Register.....	199
5.5.96 PID0 Register.....	200
5.5.97 PID1 Register.....	200
5.5.98 PID2 Register.....	201
5.5.99 PID3 Register.....	201
5.5.100 CID0 Register.....	202
5.5.101 CID1 Register.....	203
5.5.102 CID2 Register.....	203
5.5.103 CID3 Register.....	204
5.6 APB system registers.....	204
5.6.1 APB system register summary.....	204
5.6.2 SYS_ID Register.....	205
5.6.3 SYS_SW Register.....	206
5.6.4 SYS_LED Register.....	207
5.6.5 SYS_100HZ Register.....	207
5.6.6 SYS_FLAG Registers.....	208
5.6.7 SYS_CFGSW Register.....	209
5.6.8 SYS_24MHZ Register.....	209
5.6.9 SYS_PCIE_CNTL Register.....	210

5.6.10 SYS_PCIE_GBE Register.....	210
5.6.11 SYS_PROC_ID0 Register.....	211
5.6.12 SYS_FAN_SPEED Register.....	211
5.6.13 SP810_CTRL Register.....	212
5.7 Display clock and power control logic registers.....	213
5.7.1 Display clock and power control logic registers summary.....	213
5.7.2 DISPLAYCLK_CTRL Register.....	214
5.7.3 DISPLAYCLK_DIV1 Register.....	216
5.7.4 DISPLAYCLK_DIV2 Register.....	216
5.7.5 CLKFORCE_STATUS Register.....	217
5.7.6 CLKFORCE_SET Register.....	218
5.7.7 CLKFORCE_CLR Register.....	218
5.7.8 PIK_POWER_CONTROL_LOGIC Register.....	219
5.7.9 PID4 Register.....	219
5.7.10 PID0 Register.....	220
5.7.11 PID1 Register.....	220
5.7.12 PID2 Register.....	221
5.7.13 PID3 Register.....	222
5.7.14 Component ID0 Register.....	222
5.7.15 Component ID1 Register.....	223
5.7.16 Component ID2 Register.....	223
5.7.17 Component ID3 Register.....	224
5.8 GPU clock and power control logic registers.....	224
5.8.1 GPU clock and power control logic registers summary.....	224
5.8.2 GPUCLK_CTRL Register.....	226
5.8.3 GPUCLK_DIV1 Register.....	227
5.8.4 GPUCLK_DIV2 Register.....	228
5.8.5 ACLKGPU_CTRL Register.....	228
5.8.6 CLKFORCE_STATUS Register.....	229
5.8.7 CLKFORCE_SET Register.....	230
5.8.8 CLKFORCE_CLR Register.....	230
5.8.9 CAP Register.....	231
5.8.10 PIK_POWER_CONTROL_LOGIC Register.....	232
5.8.11 PID4 Register.....	232
5.8.12 PID0 Register.....	233
5.8.13 PID1 Register.....	234

5.8.14 PID2 Register.....	234
5.8.15 PID3 Register.....	235
5.8.16 Component ID0 Register.....	235
5.8.17 Component ID1 Register.....	236
5.8.18 Component ID2 Register.....	236
5.8.19 Component ID3 Register.....	237
5.9 Generic Timer registers.....	237
5.9.1 Generic Timer registers summary.....	238
5.9.2 MST_GCNT_SYNC_CTRL Register.....	238
5.9.3 SLVCHIP_GCNT_SYNC_CTRL Register.....	239
5.9.4 SLVCHIP_GCNT_INT_STATUS Register.....	239
5.9.5 SLVCHIP_GCNT_RETRY_CNT Register.....	240
5.10 APB energy meter registers.....	240
5.10.1 APB energy meter registers summary.....	241
5.10.2 SYS_I_SYS Register.....	242
5.10.3 SYS_I_CL0 Register.....	243
5.10.4 SYS_I_PCIE Register.....	244
5.10.5 SYS_I_CL1 Register.....	244
5.10.6 SYS_V_SYS Register.....	245
5.10.7 SYS_V_CL0 Register.....	246
5.10.8 SYS_V_PCIE Register.....	246
5.10.9 SYS_V_CL1 Register.....	247
5.10.10 SYS_POW_SYS Register.....	248
5.10.11 SYS_POW_CL0 Register.....	248
5.10.12 SYS_POW_PCIE Register.....	249
5.10.13 SYS_POW_CL1 Register.....	250
5.10.14 SYS_ENM_SYS Register.....	251
5.10.15 SYS_ENM_CL0 Register.....	251
5.10.16 SYS_ENM_PCIE Register.....	252
5.10.17 SYS_ENM_CL1 Register.....	253
5.10.18 SYS_I_DDR0 Register.....	254
5.10.19 SYS_I_DDR1 Register.....	255
5.10.20 SYS_V_DDR0 Register.....	255
5.10.21 SYS_V_DDR1 Register.....	256
5.10.22 SYS_POW_DDR0 Register.....	256
5.10.23 SYS_POW_DDR1 Register.....	257

5.10.24 SYS_ENM_DDR0 Register.....	258
5.10.25 SYS_ENM_DDR1 Register.....	259
5.11 UART memory addresses and control registers.....	259
A Signal descriptions.....	262
A.1 Morello-SoC JTAG connector.....	262
A.2 Trace connector.....	262
A.3 Front panel I/O header.....	264
A.4 PCI Express and CCIX slots.....	264
A.5 Power connectors.....	265
B Rainier clusters.....	266
B.1 About Rainier.....	266
B.2 Rainier features and components.....	267
B.3 Main ID Register, MIDR_EL1.....	271
C CMN-Skeena.....	272
C.1 About CMN-Skeena coherent mesh network.....	272
D DMC-Bing.....	275
D.1 About the Bing dynamic memory controller.....	275
D.2 Bing system registers.....	276
E Revisions.....	282

1 Introduction

1.1 Product revision status

The r_xp_y identifier indicates the revision status of the product described in this manual, for example, $r1p2$, where:

- r_x** Identifies the major revision of the product, for example, $r1$.
- p_y** Identifies the minor revision or modification status of the product, for example, $p2$.

1.2 Intended audience

This book is written for experienced hardware and software researchers and developers. It enables the industrial evaluation of CHERI-extended hardware.

1.3 Conventions

The following subsections describe conventions used in Arm documents.

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: developer.arm.com/glossary.

Typographic conventions

Arm documentation uses typographical conventions to convey specific meaning.

Convention	Use
<i>italic</i>	Citations.
bold	Interface elements, such as menu names. Signal names. Terms in descriptive lists, where appropriate.
monospace	Text that you can enter at the keyboard, such as commands, file and program names, and source code.
monospace bold	Language keywords when used outside example code.
monospace <u>underline</u>	A permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.

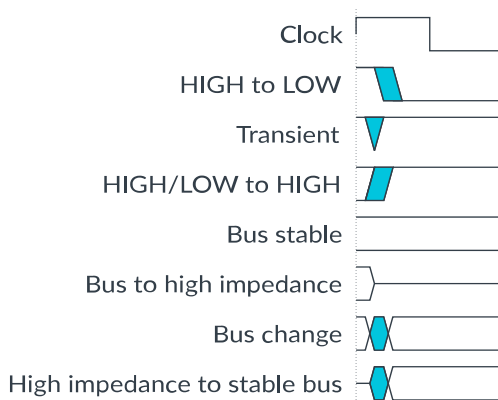
Convention	Use
<and>	Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example: <pre>MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2></pre>
SMALL CAPITALS	Terms that have specific technical meanings as defined in the <i>Arm® Glossary</i> . For example, IMPLEMENTATION DEFINED , IMPLEMENTATION SPECIFIC , UNKNOWN , and UNPREDICTABLE .
 Caution	Recommendations. Not following these recommendations might lead to system failure or damage.
 Warning	Requirements for the system. Not following these requirements might result in system failure or damage.
 Danger	Requirements for the system. Not following these requirements will result in system failure or damage.
 Note	An important piece of information that needs your attention.
 Tip	A useful tip that might make it easier, better or faster to perform a task.
 Remember	A reminder of something important that relates to the information you are reading.

Timing diagrams

The following figure explains the components used in timing diagrams. Variations, when they occur, have clear labels. You must not assume any timing information that is not explicit in the diagrams.

Shaded bus and signal areas are undefined, so the bus or signal can assume any value within the shaded area at that time. The actual level is unimportant and does not affect normal operation.

Figure 1-1: Key to timing diagram conventions



Signals

The signal conventions are:

Signal level

The level of an asserted signal depends on whether the signal is active-HIGH or active-LOW. Asserted means:

- HIGH for active-HIGH signals.
- LOW for active-LOW signals.

Lowercase n

At the start or end of a signal name, n denotes an active-LOW signal.

1.4 Additional reading

This document contains information that is specific to this product. See the following documents for other relevant information:

Table 1-2: Arm publications

Document Name	Document ID	Licensee only
AMBA® 5 CHI Architecture Specification	IHI 0050E	No
Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture	DDI 0606	No
Arm® CoreLink™ CMN-600 Coherent Mesh Network Technical Reference Manual	100180	No
Arm® CoreLink™ DMC-620 Dynamic Memory Controller Technical Reference Manual	100568	No
Arm® CoreLink™ GIC-400 Generic Interrupt Controller Technical Reference Manual	DDI 0471	No
Arm® CoreLink™ GIC-600 Generic Interrupt Controller Technical Reference Manual	100336	No
Arm® CoreLink™ MMU-600 System Memory Management Unit Technical Reference Manual	100310	No
Arm® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual	DDI 0475	No
Arm® CoreLink™ TLX-400 Network Interconnect Thin Links Supplement to Arm® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual	DSU 0028	No
Arm® CoreSight™ Components Technical Reference Manual	DDI 0314	No
Arm® Cortex®-M7 Processor Technical Reference Manual	DDI 0489	No
Arm® Development Studio Morello Edition Getting Started Guide	102233	No
Arm® DS-5 Arm DSTREAM User Guide	100955	No

Document Name	Document ID	Licensee only
Arm® DS-5 Debugger User Guide	100953	No
Arm® DS-5 Getting Started Guide	100950	No
Arm® Dual-Timer Module (SP804) Technical Reference Manual	DDI 0271	No
Arm® DynamIQ™ Shared Unit Technical Reference Manual	100453	Yes
Arm® Keil® ULINK® User's Guide	KUI 0041A	No
Arm® Neoverse™ N1 System Development Platform Technical Reference Manual	101489	No
Arm® Power Policy Unit Architecture Specification	DEN 0051E	No
Arm® PrimeCell General Purpose Input/Output (PL061) Technical Reference Manual	DDI 0190	No
Arm® PrimeCell Real Time Clock (PL031) Technical Reference Manual	DDI 0224	No
Arm® PrimeCell System Controller SP810 Technical Reference Manual	DDI 0254	No
Arm® PrimeCell UART(PL011) Technical Reference Manual	DDI 0183	No
Arm® Watchdog Module (SP805) Technical Reference Manual	DDI 0270	No
Morello Platform Model Reference Guide Version 1.3	102225	No



Note

Arm tests its PDFs only in Adobe Acrobat and Acrobat Reader. Arm cannot guarantee the quality of its documents when used with any other PDF reader.

Adobe PDF reader products can be downloaded at <http://www.adobe.com>

Other information

See the Arm website for other relevant information.

- [Arm® Developer.](#)
- [Arm® Documentation.](#)
- [Technical Support.](#)
- [Arm® Glossary.](#)

2 Overview

The Morello *System Development Platform* (SDP) is a development platform for hardware prototyping, software development, system validation, and performance profiling or tuning. It consists of the SDP prototype development board with the Morello *System-on-Chip* (SoC) running an open-source software stack. The Morello SDP board is also known as the Morello evaluation board.

The SDP serves as the *Digital Security by Design* (DSbD) technology platform prototype for the Morello architecture. This architecture introduces capabilities defined in the *Capability Hardware Enhanced RISC Instructions* (CHERI) model. Capabilities are introduced to the Armv8-A architecture profile as an extension of the Armv8 AArch64 state, with the principles proposed in version 8 of the CHERI *Instruction-Set Architecture* (ISA), to provide hardware support for fine grain protection, and building blocks for secure, scalable compartmentalization. For further information on the A-profile capability architecture, see the *Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture*.

2.1 About the Morello SDP

The Morello SDP SoC contains:

- Two CHERI-extended dual-core, out-of-order superscalar ARMv8-A-based Rainier clusters
- CHERI-extended CMN-Skeena coherent interconnect
- Two CHERI-extended DMC-Bing memory controllers
- Arm Mali™-G76 *Graphics Processing Unit* (GPU)
- Arm Mali™ D32 *Display Processing Unit* (DPU)

CMN-Skeena, a coherent memory interconnect, carries capability tag bits. Morello has two DMC-Bing memory controllers, which support memory tagging in the following implementations:

- Client mode:
 - A portion of the main memory is reserved for tag storage
 - A tag cache is used to accelerate the tag access
- Server mode: *Error Correction Code* (ECC) memory holds memory tags.

For further information, see [CMN-Skeena](#) and [DMC-Bing](#).

The purpose of Morello SDP is to:

- Target long-standing cyber-security vulnerabilities
- Enable industrial evaluation of the CHERI-extended hardware
- Enable software development
- Drive a step change for industry

- Gather evidence for adoption
- Support ongoing research and development into the effectiveness of the Capabilities protection model.

2.2 The Morello SDP at a glance

The following figures show the PC tower back panel and front panel, and the Morello board.

Figure 2-1: Back panel

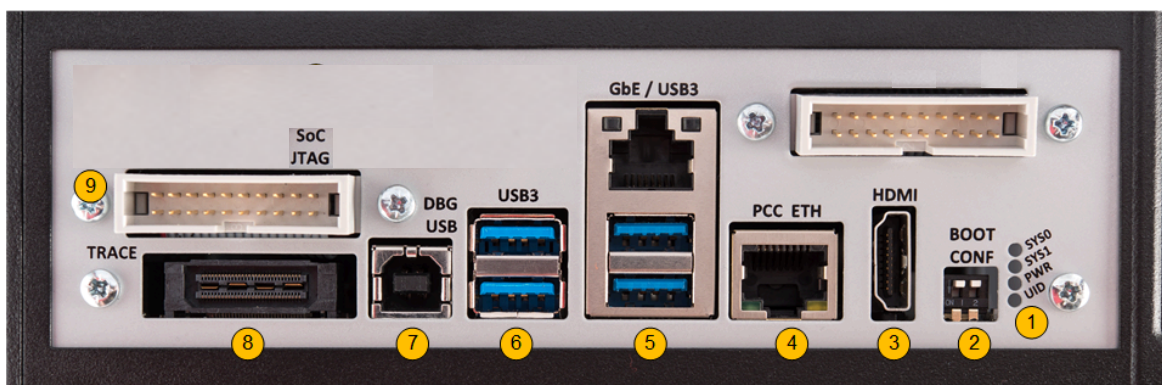
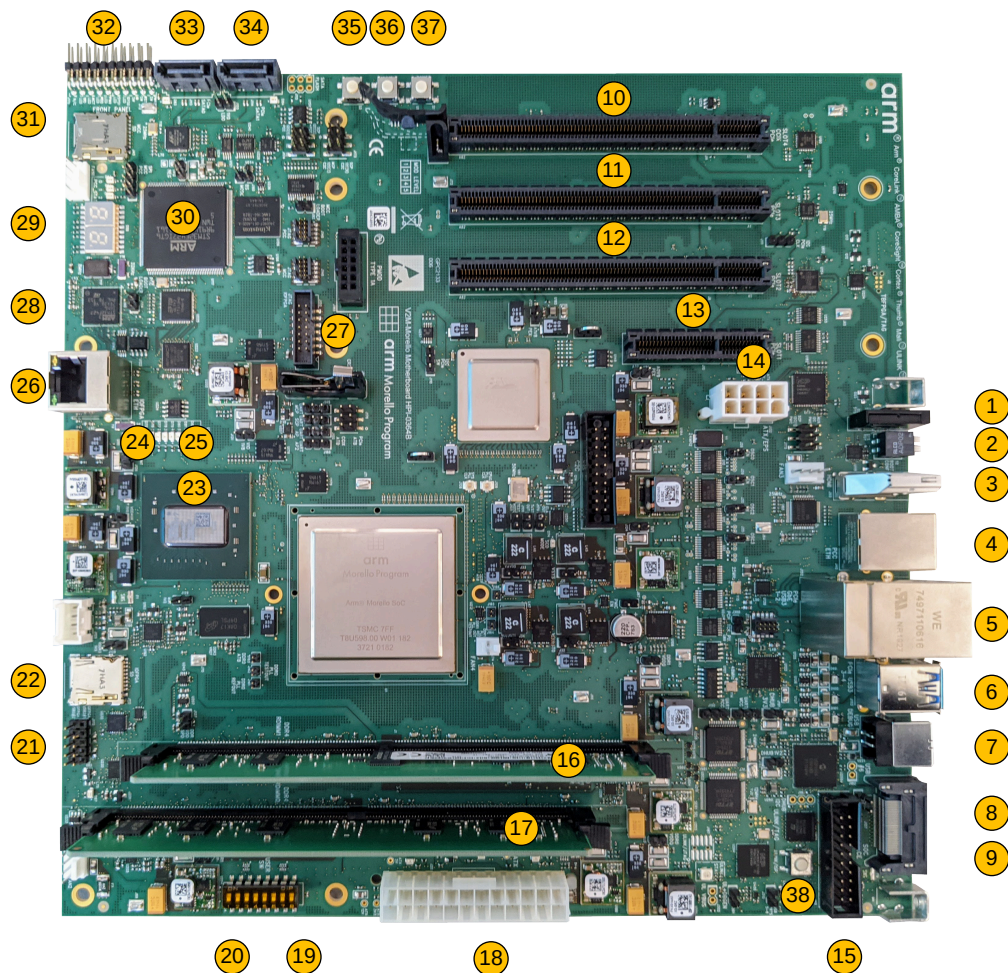


Figure 2-2: Front panel reset buttons



Figure 2-3: Morello development board



The following table describes the components, connectors, and push buttons.

Table 2-1: Key to figures

Component number	Component name	Access	Comment
1	System LEDs	Back panel.	-
2	Configuration switches	Back panel.	-
3	HDMI port	Back panel.	-
4	PCC Ethernet port	Back panel.	-
5	GbE port. USB 3.0 ports.	Back panel.	-
6	USB 3.0 ports.	Back panel.	-
7	DBG USB port	Back panel.	-
8	Morello SoC trace port	Back panel.	-

Component number	Component name	Access	Comment
9	Morello SoC JTAG port	Back panel.	Ribbon cable to SoC JTAG port, connector 15, on board.
10	Slot 4: PCIe, CCIX, ×16 connector.	Board. Remove side panel for access.	16 lanes used. Gen 4 link.
11	Slot 3: PCIe ×16 connector.	Board. Remove side panel for access.	8 lanes used, 8 lanes unused. Gen 3 link.
12	Slot 2: PCIe ×16 connector.	Board. Remove side panel for access.	16 lanes used. Gen 3 link.
13	Slot 1: PCIe ×4 connector.	Board. Remove side panel for access.	1 lane used. 3 lanes unused. Gen 3 link.
14	ATX/EPS connector	Board. Remove side panel for access.	-
15	Morello SoC JTAG port	Board.	Ribbon cable to SoC JTAG port, connector 9, on back panel.
16	RDIMM1 memory	Board. Remove side panel for access.	-
17	RDIMM0 memory	"	-
18	ATX power connector and power indicator LEDs	"	-
19	User switch SW8	"	-
20	User switch SW1	"	-
21	Reserved for use by Arm	"	-
22	IOFPGA microSD slot	"	-
23	IOFPGA	"	-
24	User LED0	"	-
25	User LED7	"	-
26	Reserved for use by Arm	"	-
27	IOFPGA JTAG	"	-
28	Platform Controller Chip (PCC)	"	-
29	7-segment display	"	-
30	Motherboard Configuration Controller (MCC)	"	-
31	MCC configuration microSD card	"	-
32	Front panel I/O connectors	"	-
33	PCIe SATA0	"	-
34	PCIe SATA1.	"	-
35	Reserved push button	-	-
36	Hardware reset button, PBRESET	Board. Remove side panel for access.	The front panel I/O connector: <ul style="list-style-type: none"> Brings the PBON and PBRESET push button functions to the front panel. Connects to the power LED.
37	ON/OFF push button, PBON	Board. Remove side panel for access.	"

Component number	Component name	Access	Comment
38	J310	"	Disables the last four USB ports.
39	ON/OFF push button, PBON, and power LED.	Front panel	"
40	Hardware reset button, PBRESET	Front panel	"

2.3 Measures to prevent damage and ensure safety

This section describes precautions that ensure safety and prevent damage to your Morello SDP board.

2.3.1 Ensuring safety

The Morello SDP is supplied in a mains-powered standard PC tower. A standard 5V ATX supply powers the board.



- Do not use the Morello SDP near equipment that is sensitive to electromagnetic emissions, for example, medical equipment.
- To reduce the risk of injury, ensure that the Morello SDP is powered down and that the fans have stopped turning before opening the chassis.

2.3.2 Operating temperature

The Morello SDP has been tested in the temperature range 0°C-40°C.

2.3.3 Preventing damage

The Morello SDP board is intended for use within a laboratory or engineering development environment.



If you remove the Morello SDP board from the PC tower, observe the following precautions:

- Never subject the board to high electrostatic potentials. Observe *ElectroStatic Discharge* (ESD) precautions when handling any board.
- Always wear a grounding strap when handling the board.
- Only hold the board by the edges.
- Avoid touching the component pins or any other metallic element.

2.4 Getting started

The Morello SDP is controlled from a serial terminal that you connect to the DBG USB port. A set of files in the non-volatile Motherboard Configuration Controller (MCC) configuration microSD card configures the board. The configuration microSD card is accessible through the DBG USB port.

The board is factory-programmed with the MCC and Platform Controller Chip (PCC), System Control Processor (SCP), Manageability Control Processor (MCP), and Application Processor (AP) firmware.

Powering up into the operating state

The minimum actions to boot the Morello SDP are as follows:

1. Connect a serial terminal to the DBG USB port on the back panel. The serial port settings must be:
 - 115.2kBaud.
 - 8N1.
 - No hardware or software flow control.

By default, the eight COM ports are connected to the following devices:

- COM<n> *Motherboard Configuration Controller (MCC).*
 - COM<n+1> *Platform Controller Chip (PCC).*
 - COM<n+2> *Application Processor (AP) 0.*
 - COM<n+3> *System Control Processor (SCP).*
 - COM<n+4> *Manageability Control Processor (MCP).*
 - COM<n+5> *Field Programmable Gate Array (FPGA) 0.*
 - COM<n+6> *Field Programmable Gate Array (FPGA) 1.*
 - COM<n+7> *Application Processor (AP) 2.*
2. Turn the mains power switch on the PC tower ON. The MCC window command prompt is shown and the system is now in the standby state. Ensure that both configuration switches on the back panel are in the OFF (up) position. See [The Morello SDP at a glance](#) for the location of the configuration switches.
 3. To complete the powerup sequence from the standby state, briefly press the PBON button. The system is now fully-powered and in the operating state.



The J310 on the motherboard can be used to reduce the number of interfaces advertised on the DBG USB interface. Adding a jumper to the J310 will reduce the number of USB interfaces by one and disable the MCP, FPGA-0, FPGA-1 and AP-2 UARTs.

Editing configuration files

The configuration microSD card contains the system configuration files. To modify the system default settings, edit or replace configuration files while the system is in standby state:

1. Ensure that the serial terminal is connected to the DBG USB port on the back panel.
2. Turn the power switch ON. The MCC window command prompt is shown and the system is now in the standby state. Ensure that both configuration switches are in the OFF (up) position.
3. Issue the following command at the MCC command prompt on the serial terminal:

- `Cmd\> usb_on`

The serial terminal now recognizes the configuration microSD card as a USB Mass Storage Device (USBMSD).

4. Edit the existing configuration files, or Drag and Drop new files.
5. Perform a Hardware Reset by pressing the PBRESET button. The system is now in the standby state.
6. Briefly press the PBON button. The system is now fully powered and in the operating state.



See [Configuration files](#) for information about the configuration files.

2.5 Accessing the ATX power cables

The Morello SDP PC tower provides SATA and other ATX power cables that you can use to connect to external hard drives. The power cables are accessed by removing the metal side panel.



Before accessing the ATX power cables, ensure that the unit is disconnected from the mains power supply.

Access the ATX power cables

To access the ATX power cables, you must gain access to the chassis:

1. Remove the large metal side panel:
2. Undo the thumbscrews at the rear of the tower.
3. Slide the side panel away from the tower.
4. The ATX power cables are now accessible, folded up inside the tower. Unfold the power cables to connect them to external hard drives.

3 Hardware description

This chapter describes the Morello SDP hardware.

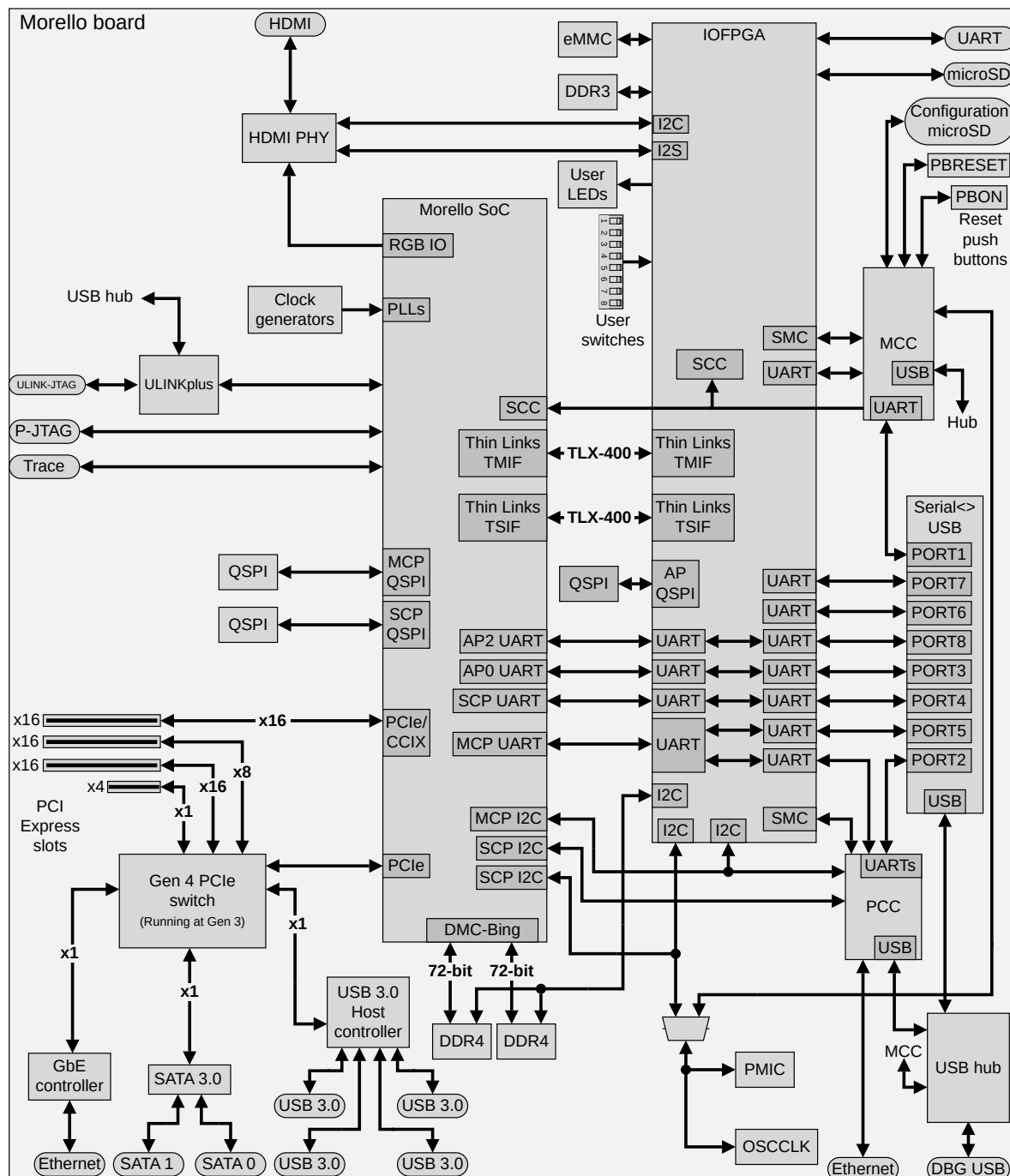
3.1 Morello SDP hardware

The support logic and peripheral interfaces of the Morello SDP support access to the Morello SoC.

Overview of the Morello SDP hardware

The following figure shows a high-level view of the system architecture.

Figure 3-1: Morello System Development Platform architecture





Note

The figure shows the default UART and USB connectivity between the components on the Morello board. The UART system is configurable using the settings in the `config.txt` file on the configuration microSD card. See the following for more information:

- [UARTs](#)
- [config.txt board configuration file](#)

Components and systems of the Morello SDP

The Morello SDP contains the following components and systems.

- *Motherboard Configuration Controller (MCC):*
 - Cortex®-M4 based controller.
 - Controls board powerup, reset, and configuration process.
 - Controls IOFPGA configuration.
 - Enables drag and drop configuration using the DBG USB connector.
 - Always powered up.
 - Static Memory Bus (SMB) connection to the IOFPGA.
 - Reads temperature measurements from the ambient sensors and controls cooling fans.
- *Platform Controller Chip (PCC):*
 - Cortex®-M4 based controller.
 - Board and SoC management.
 - Always powered up.
 - *Static Memory Bus (SMB)* connection to the IOFPGA.
 - Dual 7-segment LED display.
- *System Control Processor (SCP)* QSPI memory.
- *Manageability Control Processor (MCP)* QSPI memory.
- *Application Processor (AP)* QSPI memory.
- DDR4 memory.
- *Cache-Coherent Interconnect for Accelerators (CCIX)/PCI Express* ×16 Gen 4 slot:
 - Connects to the Gen 4 root complex and PHY on the Morello SoC.
- *PCI Express* 48-lane, 18-port, Gen 3 switch:
 - Gigabit Ethernet controller, ×1 Gen 1 link to PCIe switch.
 - SATA 3.0 controller, ×1 Gen 2 link to PCIe switch.
 - USB 3.0 controller, ×1 Gen 2 link to PCIe switch.
 - ×16 PCIe Gen 3 slot.
 - ×8 PCIe Gen 3 slot.
 - ×1 PCIe Gen 3 slot.

- Four USB 3.0 ports from USB 3.0 controller.
- Two SATA 3.0 ports from SATA 3.0 controller.
- USB hub with USB Type-B port for system configuration and Embedded ULINKplus™ operation.
- IOFPGA:
 - Low-bandwidth peripherals.
 - AXI Thin Links Master (TMIF) and Slave (TSIF) interface to the Morello SoC.
 - APB energy meter registers, for processor voltage control and current monitoring.
 - I²S audio, with output to HDMI transmitter.
- IOFPGA connections to Morello board:
 - DDR3 memory.
 - eMMC, provides user boot memory image storage.
 - microSD card, provides user boot memory image storage.
 - Two UART ports (PL011).
 - System registers, Watchdog and Real Time Clock.
 - Eight user DIP switches.
 - Eight user LEDs.
 - Definable 8-bit Type 1A PMOD connection to a PCB header.
- HDMI port:
 - On-board PHY supporting 24-bit video and direct connection to SoC.
 - UXGA 1600 x 1200 resolution.
 - Digital 8:8:8 RGB output.
- Two reset push buttons:
 - ON/OFF button, PBON.
 - Hardware reset button, PBRESET.
- Programmable oscillators.
- Embedded ULINKplus™ debugger.
- JTAG debug port.
- 32-bit Trace port.

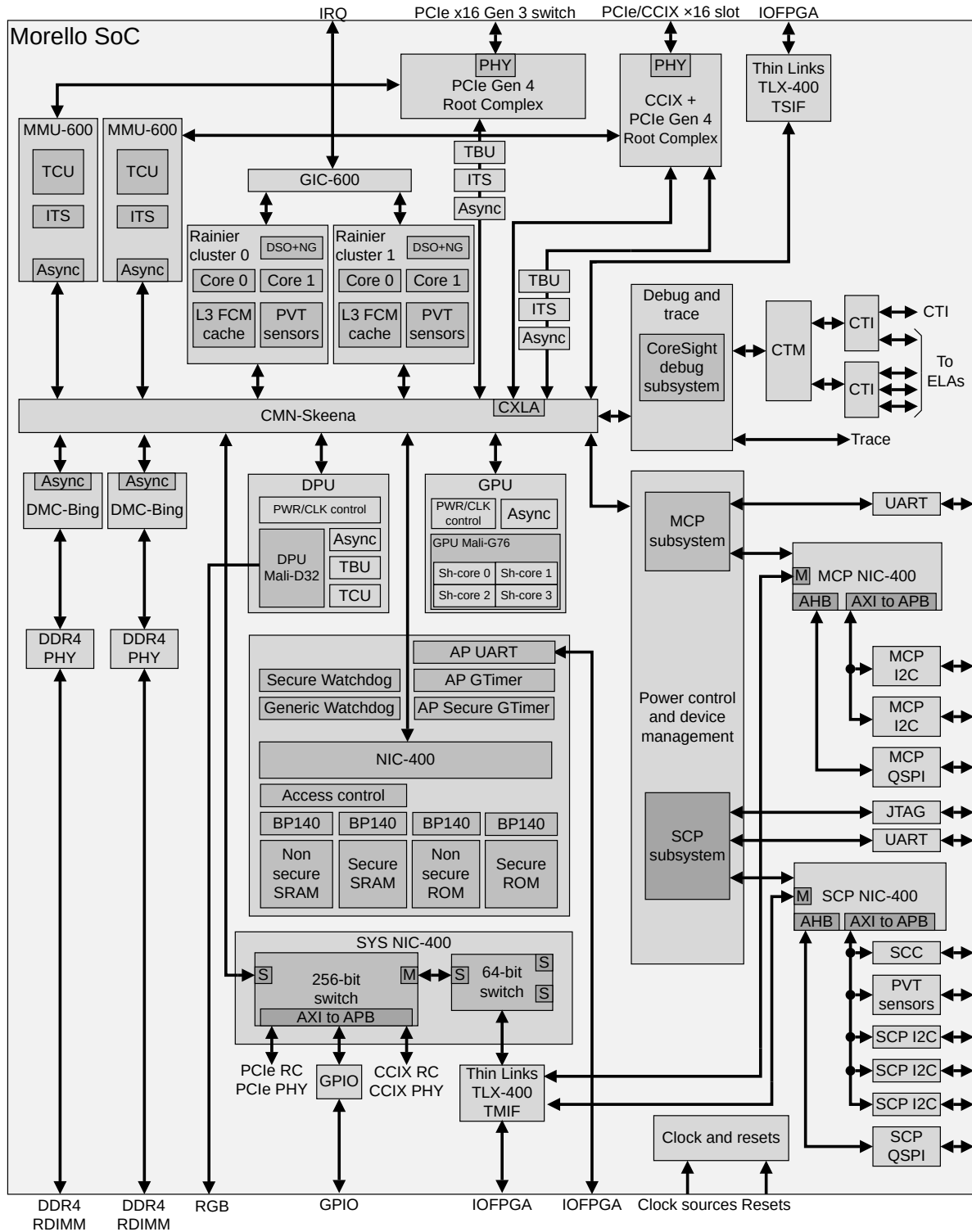
Related information

[The Morello SDP at a glance](#) on page 19

3.2 Major components of the Morello SoC

The following figure shows a high-level view of architecture of the Morello SoC.

Figure 3-2: Morello SoC



The Morello SoC contains the following components and interfaces:

- Two dual-core Rainier clusters. Each cluster has:
 - 64KB private L1 instruction cache for each core
 - 64KB private L1 data cache for each core
 - 1MB private L2 unified cache for each core
 - 1MB shared L3 unified cache in the DynamIQ™ Shared Unit (DSU) Flash Cache Module (FCM)
 - *Digital Storage Oscilloscope* (DSO) for voltage sensing and logic monitoring, with current stimulus generated by a separate *Noise Generator* (NG)
- CMN-Skeena interconnect with *Coherent Multichip Link* (CML):
 - Runs from **INTPLLCLK** default 1.85GHz
 - 1GHz clock, **CXSCLK** for CCIX block in CMN-Skeena interconnect



Note

Arm recommends that you set the CMN-Skeena clock, **INTPLLCLK**, to 1.85GHz maximum using the SCC registers. See [Clock programming and control](#), [INT_PLL_CTRL0 Register](#), and [INT_PLL_CTRL1 Register](#).

- *Embedded Logic Analyzer* (ELA) on the Rainier cores and FCM DSU
- Base element:
 - Secure region: 512KB RAM, 128KB ROM
 - Non-secure region: 64KB RAM, 4KB ROM
- GIC-600 (GICv3)
- MMU-600 Memory Management Units
- Cortex®-M7 based internal System Control Processor (SCP) and Manageability Control Processor (MCP):
 - Secure boot, power management, and device management
- CoreSight™ debug and trace
- One *Cache-Coherent Interconnect for Accelerators* (CCIX) Gen 4 Root Complex and PHY:
 - Connects to the ×16 PCI Express slot on the board
- One PCIe Gen 4 Root Complex and PHY, running as Gen 3. Connects to the following downstream slots and peripherals through a PCI Express Gen 3 switch:
 - One ×16 PCI Express slot
 - One ×8 PCI Express slot
 - One ×1 PCI Express slot
 - One ×1 Gigabit Ethernet controller
 - One ×1 SATA 3 controller
 - One ×1 USB 3 controller

- Manager and subordinate Thin Links (TLX-400) interfaces:
 - Low speed peripherals in on-board IOFPGA
- Two 72-bit DMC-Bing DDR4 controllers:
 - Support for one 288-pin RDIMM DDR4 per interface. Up to DDR4-2933 speed
 - Support for Tag Cache
- Interfaces for AP, SCP, and MCP, routed to the *Platform Controller Chip* (PCC) on the board:
 - Eight UART (PL011) interfaces, including two IOFPGA-based UARTs
 - Three I²C for SCP and two I²C interfaces for MCP
 - Two QSPI interfaces: bootup for SCP and bootup for MCP
- 8-bit GPIO (PL061) for on-board I/O
- *Serial Configuration Controller* (SCC) interface to IOFPGA
- *Process, Voltage, and Temperature* (PVT) sensors
- 32-bit *Mobile Industry Process Interface* (MIPI-60) Trace port
- JTAG debug port
- Arm Mali™-G76 *Graphics Processing Unit* (GPU):
 - Four shader cores
 - Support enabled for *Arm Frame Buffer Compression* (AFBC)
- Arm Mali™ D32 *Display Processor Unit* (DPU):
 - Single HDMI 1.4a display output
 - Support for UXGA 1600 x 1200 resolution.

For further information about the Rainier clusters, CMN-Skeena, and DMC-Bing, see:

- [Rainier clusters](#)
- [CMN-Skeena](#)
- [DMC-Bing](#)

3.3 External power

A mains supply in the range 100-240V AC powers the Morello SDP.

3.3.1 Overview of power scheme

A standard ATX power supply unit converts the mains power to low DC voltages which power the Morello board. On-board regulators supply power to the Morello board and to the power domains of the Morello SoC.

Power LEDs indicate the active power domains. See [LEDs, switches, and buttons](#) for information on the power LEDs.

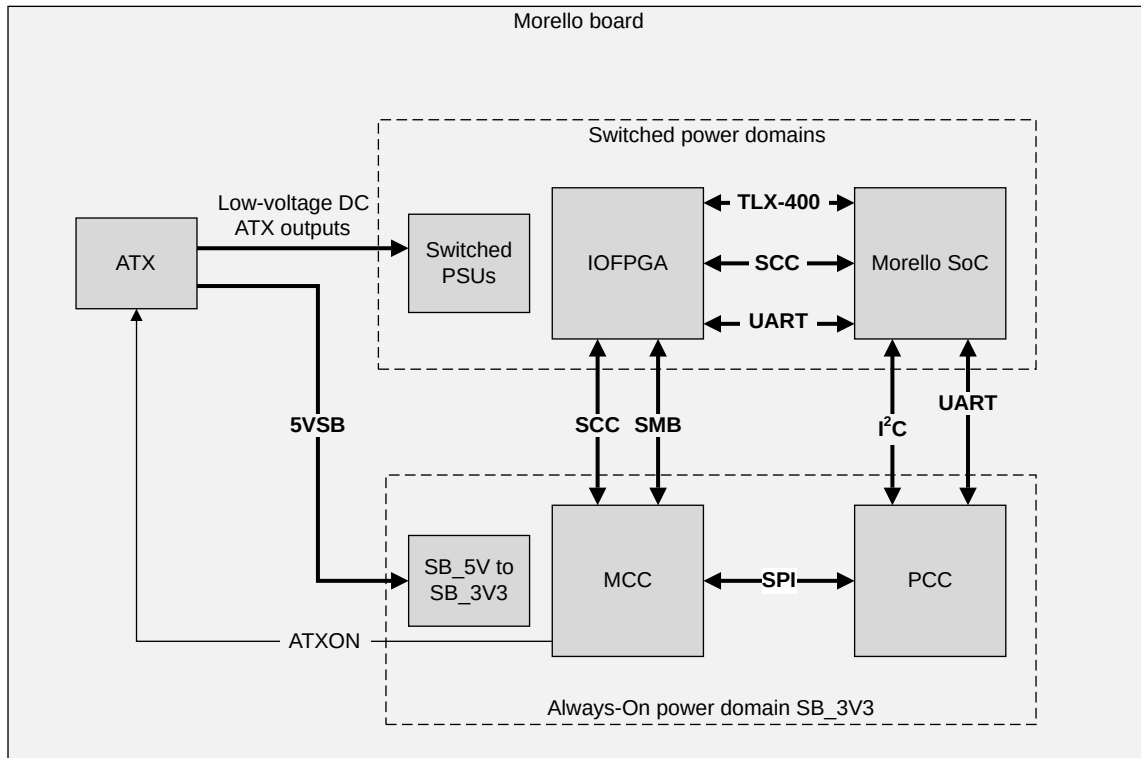
3.3.2 Power islands

The Morello board has the following power islands TEST:

- ATX Always-On region:
 - Operates in standby state.
 - Powers the *Motherboard Configuration Controller* (MCC) and *Platform Controller Chip* (PCC) subsystems only.
- Switched region:
 - Main board VIO and all other non-MCC and non-PCC supplies, including the IOFPGA.

This arrangement enables the PCC to receive a request to place the board into standby state over its local Ethernet connection. The following figure shows the Morello board power islands.

Figure 3-3: Morello board power islands



3.4 Clocks

The Morello SDP clocks drive the board and the Morello SoC.

3.4.1 Overview of clocks

Programmable clock generators on the Morello board generate clocks for the board peripherals, internal blocks in the IOFPGA, and the systems in the Morello SoC.

Phase-locked loops (PLLs) generate internal clocks in the Morello SoC for the processor clusters and other systems.

During powerup or reset, the MCC programs the clock generators according to default values defined in the `io_v0xxx.txt` configuration file. You can change the operational clock frequencies by modifying the configuration file. See [Contents of the MB directory](#) for an example `io_v0xxx.txt` configuration file.



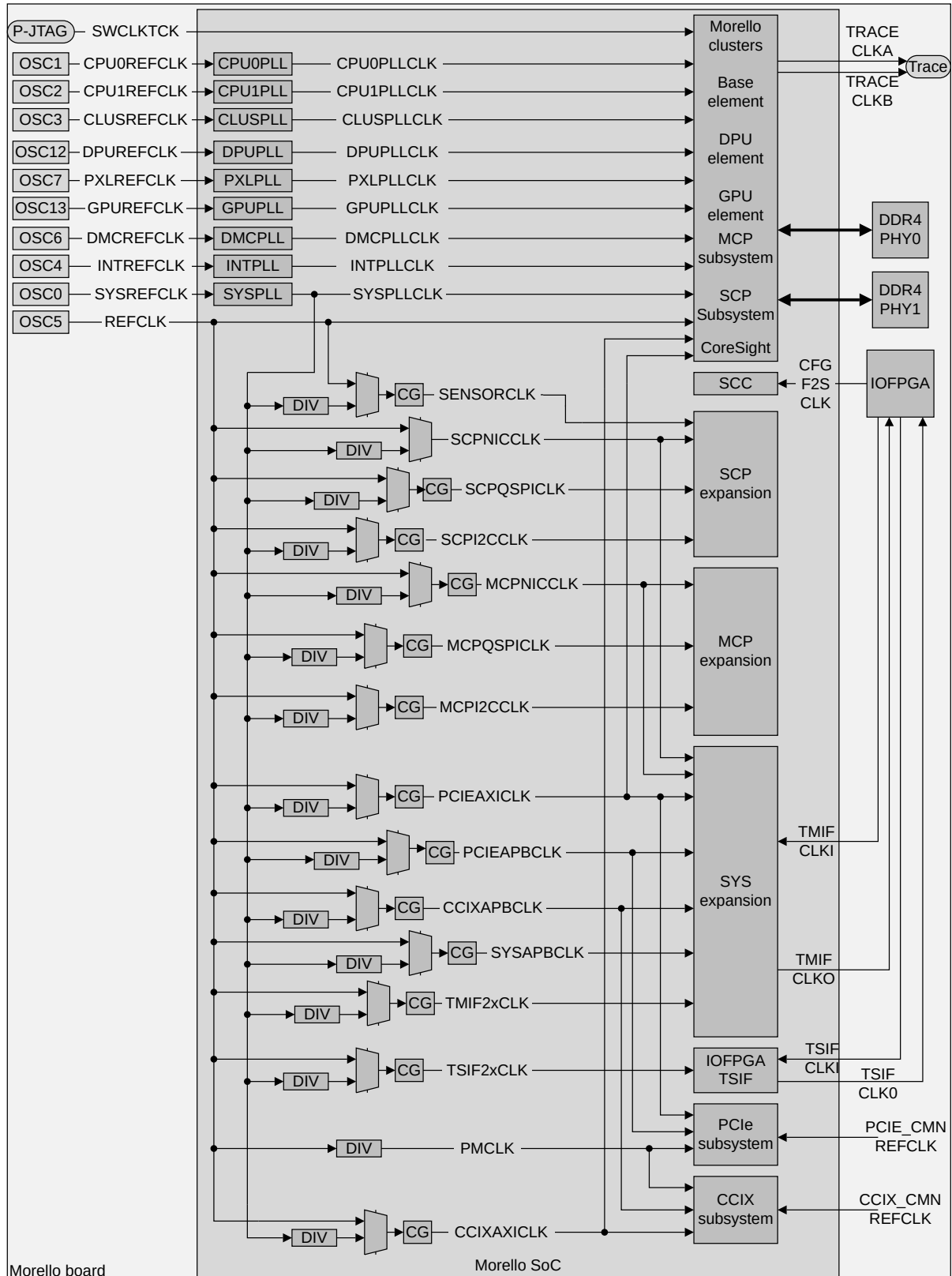
Arm recommends that you operate the Morello board at the default clock frequencies.

3.4.2 SoC clocks

Programmable clock generators on the Morello board drive PLLs in the Morello SoC which generate the internal Morello SoC clocks.

The following figure shows the programmable clock generators on the board, and the internal clocking scheme of the Morello SoC.

Figure 3-4: Test chip clocks



The *System Control Processor* (SCP) configures the PLLs, multiplexers, and dividers in the clock system during bootup.

The following table shows the Morello SoC clocks.



The default clock frequencies in this table represent an example clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in other default frequencies.

Table 3-1: Morello SoC clocks

Clock	Source	Default frequency	Description
SWCLKTCK	External debugger	50MHz	Combined P-JTAG and Serial Wire Debug (SWD) clock.
CPU0REFCLK	OSC1	50MHz	Reference clock for CPU0PLL. Generates CPU0PLLCLK for cluster 0.
CPU1REFCLK	OSC2	50MHz	Reference clock for CPU1PLL. Generates CPU1PLLCLK for cluster 1.
CLUSREFCLK	OSC3	50MHz	Reference clock for CLUSPLL. Generates CLUSPLLCLK , a common cluster clock for cluster 0 and cluster 1.
DMCREFCLK	OSC6	50MHz	Reference clock for DMCPLL. Generates DMCPLLCLK for the DDR subsystem.
DPUREFCLK	OSC12	50MHz	Reference clock for DPUPLL. Generates DPULLCLK for the DPU element.
PXLREFCLK	OSC7	50MHz	Reference clock for PXLPLL. Generates PXLPLLCLK for HDMI.
GPUREFCLK	OSC13	50MHz	Reference clock for GPUPLL. Generates GPUPLLCLK for the GPU element.
INTREFCLK	OSC4	50MHz	Reference clock for INTPLL. Generates INTPLLCLK for the CMN-600 Coherent Mesh Network.
SYSREFCLK	OSC0	50MHz	Reference clock for SYSPLL. Generates the main system clock SYSPLLCLK , and other clocks through programmable dividers.
REFCLK	OSC5	50MHz	Always-On reference clock.
CPU0PLLCLK	CPU0PLL	2.5GHz	Cluster-specific clock for cluster 0.
CPU1PLLCLK	CPU1PLL	2.5GHz	Cluster-specific clock for cluster 1.
CLUSPLLCLK	CLUSPLL	2GHz	Clock for cluster 0 and cluster 1.
DMCPLLCLK	DMCPLL	1.467GHz	Clock for DDR subsystem.
INTPLLCLK	INTPLL	1.85GHz	CMN-600 Coherent Mesh network clock.
DPULLCLK	DPUPLL	350MHz	Clock for DPU element.
DPUACLKM	DPUPLL	350MHz	Transport clock for DPU.
DPUACLKS	DPUPLL	350MHz	Transport clock for DPU.
DISPLAYCLK	DPUPLL	350MHz	DPU element divided primary mode.
PXLPLLCLK	PXLPLL	162MHz	Clock for Display unit in DPU element.
GPUPLLCLK	GPUPLL	650MHz	Clock for GPU element.
GPUACLKS	GPUPLL	650MHz	Transport clock for GPU.
GPUACLKM	GPUPLL	650MHz	Transport clock for GPU.
GPUCLK	GPUPLL	650MHz	GPU element divided primary mode.
SYSPLLCLK	SYSPLL	2.4GHz	Main system clock.

Clock	Source	Default frequency	Description
SENSORCLK	SYSPLL	100MHz	Sensor clock.
SCPNICCLK	SYSPLL	300MHz	SCP NIC-400 clock.
SCPQSPICLK	REFCLK	50MHz	SCP QSPI reference clock
SCPI2CCLK	REFCLK	100MHz	SCP I ² C clock
MCPNICCLK	SYSPLL	300MHz	MCP NIC-400 clock.
MCPQSPICLK	REFCLK	50MHz	MCP QSPI reference clock
MCPI2CCLK	REFCLK	100MHz	MCP I ² C clock
PCIEAXICLK	SYSPLL	1.2GHz	PCIe AXI clock
PCIEAPBCLK	SYSPLL	200MHz	PCIe APB clock
CCIXAPBCLK	SYSPLL	200MHz	CCIX APB clock
SYSAPBCLK	SYSPLL	120MHz	System expansion APB peripherals
TMIF2xCLK	SYSPLL	170MHz	IOFPGA TMIF 2× clock
TSIF2xCLK	SYSPLL	170MHz	IOFPGA TSIF 2× clock
CCIXAXICLK	SYSPLL	1.2GHz	CCIX AXI clock
TRACECLKA	TPIU	150MHz	Trace clocks to connector
TRACECLKB	TPIU	150MHz	Trace clocks to connector
PMCLK	REFCLK	25MHz	Power management clock for PCIe and CCIX
CFG_F2S_CLK	MCC	5MHz	<i>Serial Configuration Controller</i> (SCC) interface clock. This clock is a data strobe, not a free running clock.
TMIF_CLKI	IOFPGA TLX-400 manager interface	85MHz	Thin Links-based AXI manager interface clock received from IOFPGA with incoming data from IOFPGA.
TMIF_CLKO	Morello SoC TLX-400 manager interface	85MHz	Thin Links-based AXI manager interface clock exported from Morello SoC with data transmitted from Morello SoC to IOFPGA.
TSIF_CLKI	IOFPGA TLX-400 subordinate interface	85MHz	Thin Links-based AXI subordinate interface clock received from IOFPGA with incoming data from IOFPGA.
TSIF_CLKO	Morello SoC TLX-400 subordinate interface	85MHz	Thin Links-based AXI subordinate interface clock exported from Morello SoC with data transmitted from Morello SoC to IOFPGA.
PCIE_CMN_REFCLK	Clock buffer	100MHz	Fixed differential reference clock for PCIe U-PHY
CCIX_CMN_REFCLK	Clock buffer	100MHz	Fixed differential reference clock for CCIX U-PHY
CXCLK	CCIX PCIe PHY	1GHz	CXLA clock in CMN-600
PXLCLK	DPU	162MHz	Output clock to HDMI PHY.

3.4.3 Clock programming and control

The clocks are controlled by *Serial Configuration Control* (SCC) registers in the Morello SoC.

The SCC clock control registers select the sources, control the PLLs, and set the clock division values.

There are three classes of SCC clock control registers: PLL control registers, *_CTRL registers, and *_DIV registers.

The PLL control registers set the multiplication factor of each PLL by setting the values of REFDIV, FBDIV, and POSTDIV, where:

Output clock frequency = (Input clock frequency/REFDIV)×FBDIV/POSTDIV.

- REFDIV is input frequency division value.
- FBDIV is the PLL feedback division value.
- POSTDIV is the PLL output frequency division value.

The *_CTRL registers select the source clocks, **SYSPLLCLK** or **REFCLK**, for the internal clocks.

The *_DIV registers set the division values of the clock dividers. *_DIV registers are used when **SYSPLLCLK** is selected as the source clock.

The following table shows the SCC clock control registers.

Table 3-2: SCC clock control registers

Register	Register function	Register description
PMCLK_DIV	Sets value of divider value to generate PMCLK from REFCLK .	See PMCLK_DIV Register .
SYSAPBCLK_CTRL	Selects input clock to generate SYSAPBCLK .	See SYSAPBCLK_CTRL Register .
SYSAPBCLK_DIV	Sets value of divider value to generate SYSAPBCLK from SYSPLLCLK .	See SYSAPBCLK_DIV Register .
IOFPGA_TMIF2XCLK_CTRL	Selects input clock to generate TMIF2XCLK .	See IOFPGA_TMIF2XCLK_CTRL Register .
IOFPGA_TMIF2XCLK_DIV	Sets value of divider value to generate TMIF2XCLK from SYSPLLCLK .	See IOFPGA_TMIF2XCLK_DIV Register .
IOFPGA_TSIF2XCLK_CTRL	Selects input clock to generate TSIF2XCLK .	See IOFPGA_TSIF2XCLK_CTRL Register .
IOFPGA_TSIF2XCLK_DIV	Sets value of divider value to generate TSIF2XCLK from SYSPLLCLK .	See IOFPGA_TSIF2XCLK_DIV Register .
SCPNICCLK_CTRL	Selects input clock to generate SCPNICCLK .	See SCPNICCLK_CTRL Register .
SCPNICCLK_DIV	Sets value of divider value to generate SCPNICCLK from SYSPLLCLK .	See SCPNICCLK_DIV Register .
SCPI2CCLK_CTRL	Selects input clock to generate SCPI2CCLK .	See SCPI2CCLK_CTRL Register .
SCPI2CCLK_DIV	Sets value of divider value to generate SCPI2CCLK from SYSPLLCLK .	See SCPI2CCLK_DIV Register .
SCPQSPICLK_CTRL	Selects input clock to generate SCPQSPICLK .	See SCPQSPICLK_CTRL Register .
SCPQSPICLK_DIV	Sets value of divider value to generate SCPQSPICLK from SYSPLLCLK .	See SCPQSPICLK_DIV Register .
SENSORCLK_CTRL	Selects input clock to generate SENSORCLK .	See SENSORCLK_CTRL Register .
SENSORCLK_DIV	Sets value of divider value to generate SENSORCLK from SYSPLLCLK .	See SENSORCLK_DIV Register .
MCPNICCLK_CTRL	Selects input clock to generate MCPNICCLK .	See MCPNICCLK_CTRL Register .
MCPNICCLK_DIV	Sets value of divider value to generate MCPNICCLK from SYSPLLCLK .	See MCPNICCLK_DIV Register .
MCPI2CCLK_CTRL	Selects input clock to generate MCPI2CCLK .	See MCPI2CCLK_CTRL Register .

Register	Register function	Register description
MCPI2CCLK_DIV	Sets value of divider value to generate MCPI2CCLK from SYSPLLCLK .	See MCPI2CCLK_DIV Register .
MCPQSPICLK_CTRL	Selects input clock to generate MCPQSPICLK .	See MCPQSPICLK_CTRL Register .
MCPQSPICLK_DIV	Sets value of divider value to generate MCPQSPICLK from SYSPLLCLK .	See MCPQSPICLK_DIV Register .
PCIEAXICLK_CTRL	Selects input clock to generate PCIEAXICLK .	See PCIEAXICLK_CTRL Register .
PCIEAXICLK_DIV	Sets value of divider value to generate PCIEAXICLK from SYSPLLCLK .	See PCIEAXICLK_DIV Register .
CCIXAXICLK_CTRL	Selects input clock to generate CCIXAXICLK .	See CCIXAXICLK_CTRL Register .
CCIXAXICLK_DIV	Sets value of divider value to generate CCIXAXICLK from SYSPLLCLK .	See CCIXAXICLK_DIV Register .
PCIEAPBCLK_CTRL	Selects input clock to generate PCIEAPBCLK .	See PCIEAPBCLK_CTRL Register .
PCIEAPBCLK_DIV	Sets value of divider value to generate PCIEAPBCLK from SYSPLLCLK .	See PCIEAPBCLK_DIV Register .
CCIXAPBCLK_CTRL	Selects input clock to generate CCIXAPBCLK .	See CCIXAPBCLK_CTRL Register .
CCIXAPBCLK_DIV	Sets value of divider value to generate CCIXAPBCLK from SYSPLLCLK .	See CCIXAPBCLK_DIV Register .
SYS_CLK_EN	Enables or disables internally generated clocks.	See SYS_CLK_EN Register .
CPU0_PLL_CTRL0	Controls CPU0PLL to generate CPU0PLLCLK .	See CPU0_PLL_CTRL0 Register .
CPU0_PLL_CTRL1	Controls CPU0PLL to generate CPU0PLLCLK .	See CPU0_PLL_CTRL1 Register .
CPU1_PLL_CTRL0	Controls CPU1PLL to generate CPU1PLLCLK .	See CPU1_PLL_CTRL0 Register .
CPU1_PLL_CTRL1	Controls CPU1PLL to generate CPU1PLLCLK .	See CPU1_PLL_CTRL1 Register .
CLUS_PLL_CTRL0	Controls CLUSPLL to generate CLUSPLLCLK .	See CLUS_PLL_CTRL0 Register .
CLUS_PLL_CTRL1	Controls CLUSPLL to generate CLUSPLLCLK .	See CLUS_PLL_CTRL1 Register .
SYS_PLL_CTRL0	Controls SYSPLL to generate SYSPLLCLK .	See SYS_PLL_CTRL0 Register .
SYS_PLL_CTRL1	Controls SYSPLL to generate SYSPLLCLK .	See SYS_PLL_CTRL1 Register .
DMC_PLL_CTRL0	Controls DMCPLL to generate DMCPLLCLK .	See DMC_PLL_CTRL0 Register .
DMC_PLL_CTRL1	Controls DMCPLL to generate DMCPLLCLK .	See DMC_PLL_CTRL1 Register .
INT_PLL_CTRL0	Controls INTPLL to generate INTPLLCLK .	See INT_PLL_CTRL0 Register .
INT_PLL_CTRL1	Controls INTPLL to generate INTPLLCLK .	See INT_PLL_CTRL1 Register .
GPU_PLL_CTRL0	Controls GPUPLL to generate GPUPLLCLK .	See GPU_PLL_CTRL0 Register .
GPU_PLL_CTRL1	Controls GPUPLL to generate GPUPLLCLK .	See GPU_PLL_CTRL1 Register .
DPU_PLL_CTRL0	Controls DPUPLL to generate DPUPLLCLK .	See DPU_PLL_CTRL0 Register .
DPU_PLL_CTRL1	Controls DPUPLL to generate DPUPLLCLK .	See DPU_PLL_CTRL1 Register .
PXL_PLL_CTRL0	Controls PXLPLL to generate PXLPLLCLK .	See PXL_PLL_CTRL0 Register .
PXL_PLL_CTRL1	Controls PXLPLL to generate PXLPLLCLK .	See PXL_PLL_CTRL1 Register .

3.4.4 IOFPGA clocks

Programmable clock generators on the Morello board generate clocks for the internal systems of the IOFPGA.

The IOFPGA Thin Links interfaces generate clocks for data transmitted to the Morello SoC interfaces. The IOFPGA also generates the *Serial Configuration Controller* (SCC) clock data strobe.

The following table shows the IOFPGA clocks.

Table 3-3: IOFPGA clocks

Clock	Source	Frequency	Description
IOFPGA_ACLK	OSC9	85MHz	Boot up clock. Drives IOFPGA OSC0.
IOFPGA_TLXCLK	OSC8	80MHz	Reserved.
IOFPGA_TLX2XCLK	OSC14	170MHz	2x frequency of IOFPGA_TLXCLK .
IOFPGA_AUDCLK	OSC10	12.288MHz	Drives audio clock I2SCLK .
IOFPGA_CLK24M	FPGA_CLK24M	24MHz	Drives MMCM at 100MHz.
IOFPGA_SPARECLK	OSC11	50MHz	Spare.
S32KCLK	CLK_32K	32.768kHz	Standalone clock for Real Time Clock (RTC)
IOFPGA_DDR3_SYSCLK	GTX clock	100MHz	Drives DDR3 controller reference clock at 400MHz.
Thin Links-based AXI manager and subordinate interface clocks between IOFPGA and Morello SoC.	-	85MHz from SoC to IOFPGA. 85MHz from IOFPGA to SoC.	See SoC clocks for descriptions of Thin Links clocks.
SMBM_CLK	MCC	40MHz	SMB clock
SMBP_CLK	PCC	40MHz	SMB clock
CFG_M2F_CLK	MCC	10MHz	<i>Serial Configuration Controller</i> (SCC) interface clock. This clock is a data strobe, not a free running clock.
CFG_CLK	IOFPGA SCC interface	25MHz	SCC interface clock. This clock is a data strobe, not a free running clock.
PCIE_CMN_REFCLK	Clock buffer	100MHz	Fixed differential reference clock for PCIe U-PHY
CCIX_CMN_REFCLK	Clock buffer	100MHz	Fixed differential reference clock for CCIX U-PHY
MMB_SCK	IOFPGA	24.576MHz	Audio clock output

3.5 Resets

The Morello SDP provides reset signals for the Morello board and Morello SoC.

Morello board resets

The Morello board has the following resets.

Table 3-4: Morello board resets

Reset	Source	Target	Comment
nPBRESET	Powerup reset. Hardware reset user push button PBRESET.	<i>Motherboard Configuration Controller (MCC)</i> and entire board	Main board powerup reset and hard reset from the user hardware reset push button. The entire system goes to standby state.
UART0_DSR	External device	MCC and entire board	Remote UART reset. Must be enabled by use of configuration switch SW1. There are reset push buttons, configuration DIP switches, and user DIP switches on the Morello board.
CB_CFGnRST	MCC	<i>Platform Controller Chip (PCC)</i> and entire board	The MCC controls the PCC reset.
IOFPGA_nPOR	MCC	IOFPGA, powerup reset sections.	Enables the IOFPGA internal PLLs to be reset and become stable before release of logic reset.
IOFPGA_nRST	MCC	IOFPGA logic, internal blocks.	Resets main IOFPGA logic blocks.
nPBON	ON/OFF push button, PBON.	MCC and PCC	Push button to power up. By default, this button powers up the system in standby mode. nPBON generates an interrupt to either the MCC or PCC to control the powerup sequence.

Morello SoC resets

The Morello SoC has the following resets from the board.

Table 3-5: Morello SoC resets

Reset	Source	Comment
SOC_nPOR	MCC, IOFPGA.	Main powerup reset for the whole system except some SCC logic. De-assertion of this input initiates the powerup sequence.
SOC_nSRST	MCC/IOFPGA/ external debug unit.	Debug through reset signal. This signal enables the debug tools to debug Cortex®-M7 (SCP, MCP) and the Rainier (AP) before the processors leave reset.
nTRST	External debug unit	JTAG reset. Resets the CoreSight DAP.
CFG_nRST	MCC through IOFPGA	Reset signal for the serial interface to the System Configuration Controller (SCC).

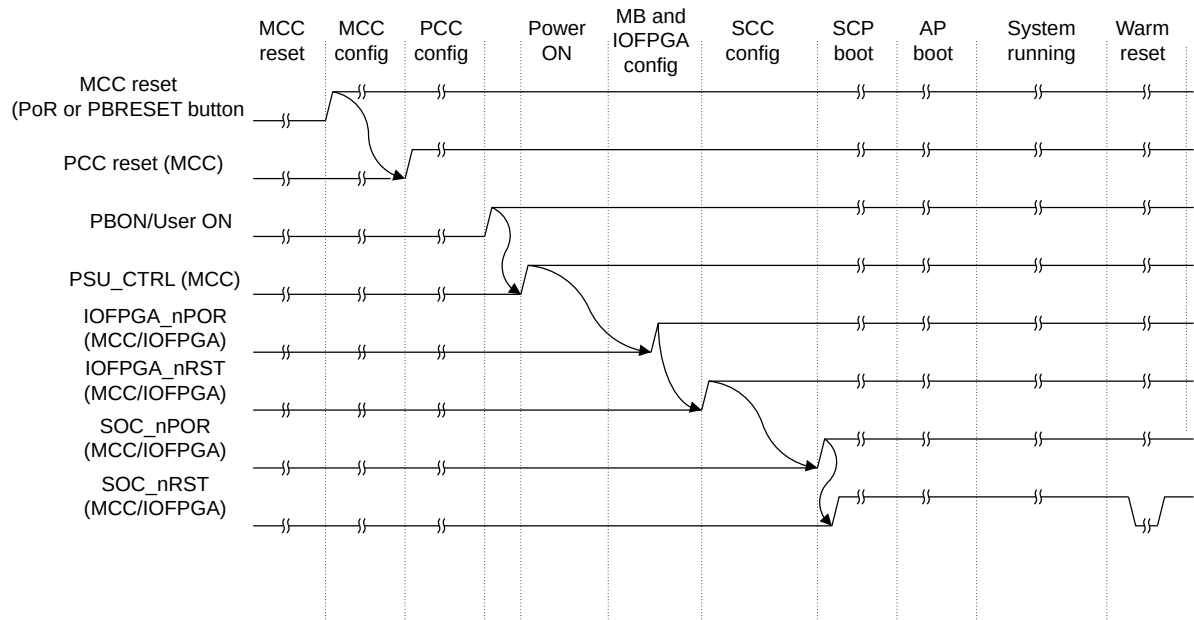
Reset sequence

The system can operate in two modes:

- Mobile mode:
 - The board and SCP are brought up by the MCC alone.
- Enterprise mode:
 - The MCC brings the board up to standby state and is then under the control of the PCC.

The following figure shows the board and SoC reset sequence.

Figure 3-5: Reset sequence



Related information

[The Morello SDP at a glance](#) on page 19

3.6 IOFPGA

The IOFPGA provides access to low-bandwidth peripherals that the Morello SoC does not provide. The Morello SoC connects to the IOFPGA through AXI Thin Links (TLX-400) manager and subordinate interfaces.

3.6.1 Overview of IOFPGA

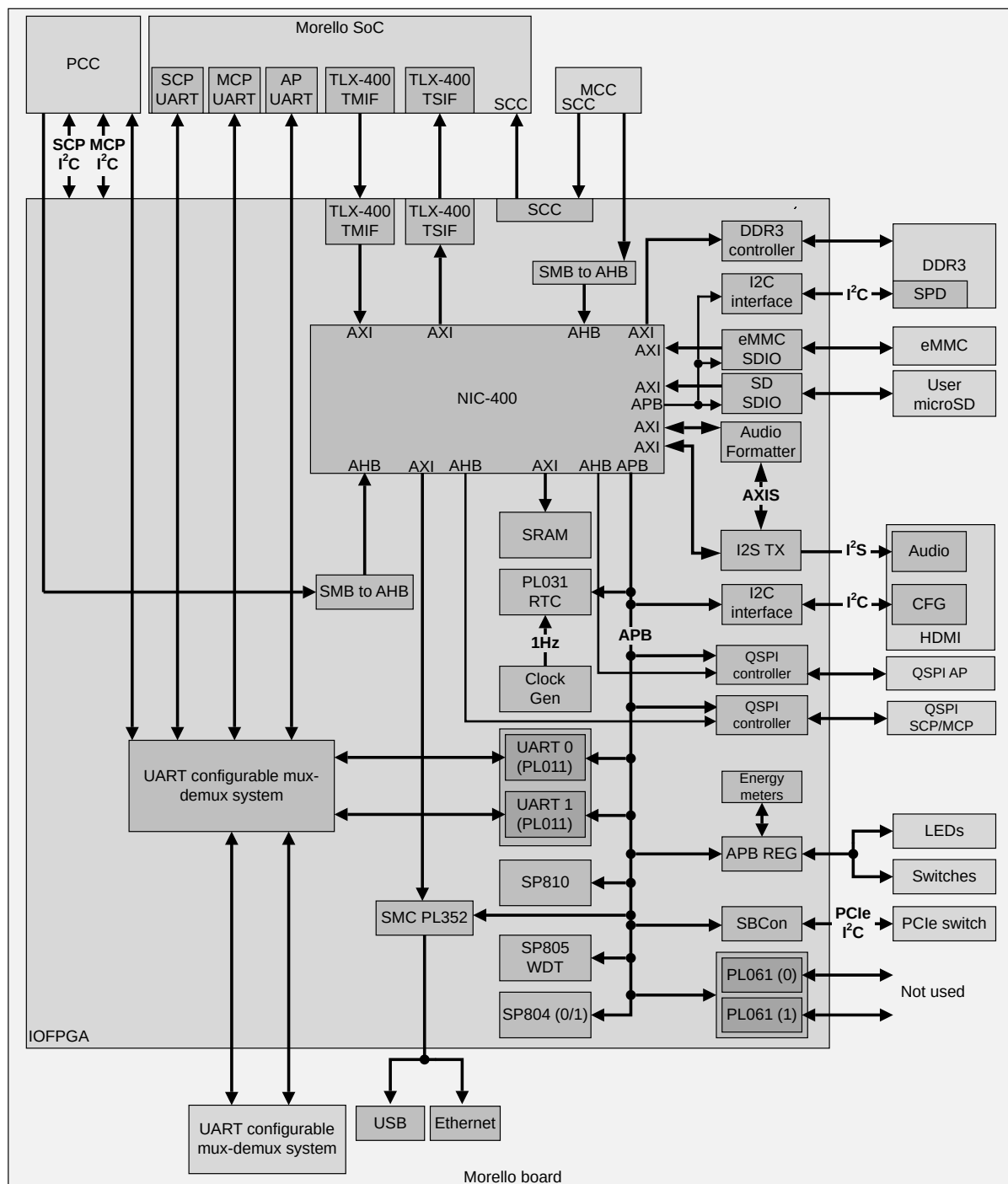
The IOFPGA on the Morello board contains the following blocks and interfaces:

- eMMC device
- microSD card controller
- QSPI controllers
- DDR3 controller
- PL031 Real Time Clock (RTC)
- SP804 Timers
- SP805 Watchdog

- SP810 System controller
- PL011 UARTs
- 1MB SRAM
- I²C configuration of PCIe switch and HDMI PHY
- I²S audio configuration of HDMI PHY
- APB mapping to user LEDs and switches
- APB-mapped energy meter registers

The following figure shows the internal architecture of the IOFPGA and its connectivity to external peripherals, the Morello SoC, the MCC, and the PCC.

Figure 3-6: IOFPGA internal architecture





Note

The UART system in the IOFPGA and Morello SoC is configurable using the settings in the `config.txt` file. See the following for information on the UART system, and on configuring the UART system.

- [UARTs](#).
- [config.txt board configuration file](#).

3.6.2 IOFPGA interrupts

There are 38 interrupts sources generated by peripherals of the IOFPGA. The IOFPGA provides twelve interrupt signals to the SoC and additional two to the *Motherboard Configuration Controller* (MCC) and *Platform Controller Chip* (PCC).

The following table shows the IOFPGA interrupt connection and muxing options.

Table 3-6: IOFPGA interrupt connection and muxing

SoC IRQ signal	IOFPGA interrupts	Muxing select
AP_EXT_INT	INT MUX 0	SCC REG 8 [7:0]
AP_EXT_ETH_INT	IOFPGA Ethernet Int	-
SOC_IRQ[0]	Audio Formatter Int	-
SOC_IRQ[1]	I ² C for HDMI Int	-
SOC_IRQ[2]	I ² S Transmitter	-
SOC_IRQ[3]	INT MUX 1	SCC REG 8 [15:8]
SOC_IRQ[4]	INT MUX 2	SCC REG 8 [23:16]
SOC_IRQ[5]	INT MUX 3	SCC REG 8 [31:24]
SOC_IRQ[6]	INT MUX 4	SCC REG 9 [7:0]
SOC_IRQ[7]	INT MUX 5	SCC REG 9 [15:8]
SCP_EXT_INT	INT MUX 6	SCC REG 9 [23:16]
MCP_EXT_INT	INT MUX 7	SCC REG 9 [31:24]
F2MCC_INT	INT MUX 8	SCC REG 10 [7:0]
F2PCC_INT	INT MUX 9	SCC REG 10 [15:8]

The following table shows the IOFPGA interrupt sources, connected to the internal multiplexer of the IOFPGA.

Table 3-7: IOFPGA interrupt sources

Interrupt ID	IRQ level	Source	Comment
IRQ[0]	1	Timer 0	-
IRQ[1]	1	Timer 1	-
IRQ[2]	1	Timer 2	-
IRQ[3]	1	Timer 3	-
IRQ[4]	1	Watchdog	-

Interrupt ID	IRQ level	Source	Comment
IRQ[5]	1	Real Time Clock	-
IRQ[6]	1	CFGINT (System)	-
IRQ[7]	1	FUNINT (System)	-
IRQ[8]	1	UART 0	-
IRQ[9]	1	UART 1	-
IRQ[10]	1	GPIO 0	-
IRQ[11]	1	GPIO 1	-
IRQ[12]	1	QSPI (AP)	-
IRQ[13]	-	QSPI (SCP/MCP)	-
IRQ[14]	-	USB	-
IRQ[15]	1	User microSD	-
IRQ[16]	1	eMMC	-
IRQ[17]	0	HDMI PHY INT	-
IRQ[18]	1	CFGINT (SCC)	-
IRQ[19]	1	MSCP_SS_RSTREQ	From SoC
IRQ[20]	1	M2F_AP_EXTINT	MCC to AP
IRQ[21]	1	M2F_SCP_EXTINT	MCC to SCP
IRQ[22]	1	M2F_MCP_EXTINT	MCC to MCP
IRQ[23]	1	P2F_AP_EXTINT	PCC to AP
IRQ[24]	1	P2F_SCP_EXTINT	PCC to SCP
IRQ[25]	1	P2F_MCP_EXTINT	PCC to MCP
IRQ[26]	1	CCIX_nWAKE	-
IRQ[27]	1	FATAL_ERRn from PCIe switch	-
IRQ[28]	1	Reserved	-
IRQ[29]	1	DDR4 EEPROM I ² C	-
IRQ[30]	0	DDR4_nEVENT0	-
IRQ[31]	0	DDR4_nEVENT1	-
IRQ[32]	0	PCIe_nWAKE	-
IRQ[33]	0	PCIe switch	-
IRQ[34]	0	IOFPGA Ethernet Int	-
IRQ[35]	1	I2S Transmitter	-
IRQ[36]	1	I2C for HDMI	-
IRQ[37]	1	Audio Formatter	-

3.7 PCI Express and CCIX systems

The Morello SDP provides PCI Express Gen 4, and *Cache-Coherent Interconnect for Accelerators* (CCIX) expansion.

3.7.1 Overview of PCIe and CCIX systems

The Morello SoC provides two PCI Express Gen 4 x16, 16Gbps, independent interfaces. One of the interfaces supports *Cache-Coherent Interconnect for Accelerators* (CCIX) technology.

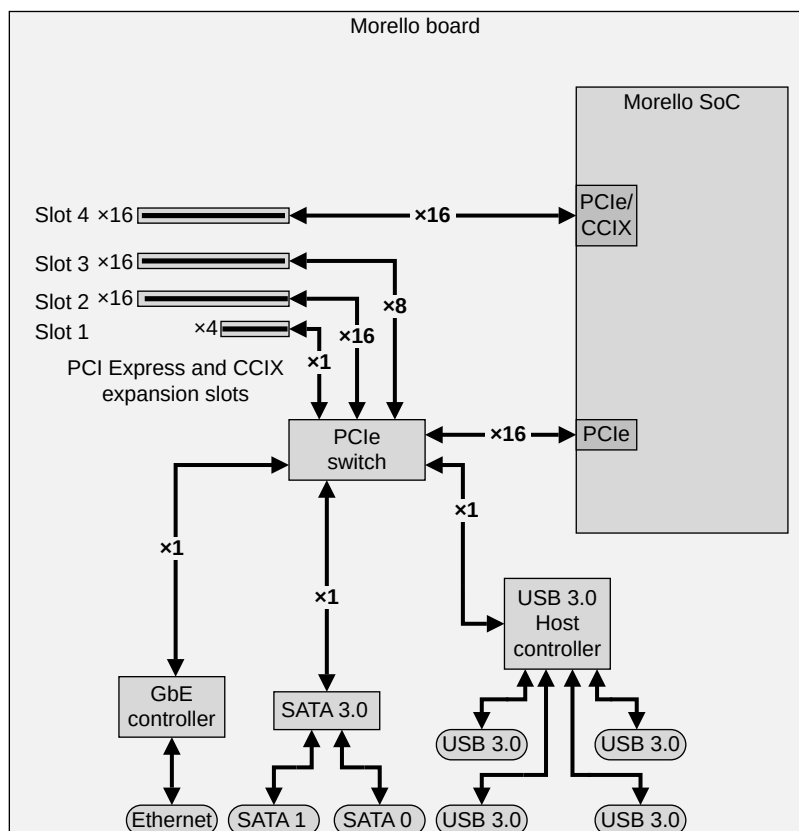
One Gen 4 Root Complex on the Morello SoC connects directly to a Gen 3 switch. Downstream of the switch provides access to three PCIe slots, a four-port USB 3.0 Host Controller, a two-port SATA controller, and one Gigabit Ethernet port.

The other Gen 4 Root Complex on the Morello SoC provides access through one x16 CCIX slot which also functions as a standard PCIe Gen 4 interface.

The GbE and USB 3.0 connectors are accessible on the back panel. The PCIe and CCIX slots, and the SATA ports, are accessible by removing the side panel.

The following figure shows the Morello SDP PCI Express and CCIX systems.

Figure 3-7: PCI Express and CCIX systems



3.7.2 PCI Express and CCIX expansion slots

The following table shows the PCIe expansion slots on the Morello board.

Table 3-8: PCI Express expansion slots

Slot number	PCIe lane connector size	Number of lanes connected	Unused lanes	Comment
Slot 1	×4	1	3	PCIe
Slot 2	×16	16	0	PCIe
Slot 3	×16	8	8	PCIe
Slot 4	×16	16	0	PCIe CCIX dual-use

- Using slot 2 as the ×16 slot:



Note

- Enables use of a double slot 150W/300W card which covers slot 3 and makes it unusable.
- Enables use of a double slot card in slot 1.
- It is not possible to simultaneously use a ×16 Triple-slot card in slot 2 and a ×16 lane CCIX in slot 3.

3.7.3 SATA 3.0 ports

The PCIe switch connects to the SATA 3.0 controller over a ×1 PCIe link.

The SATA 3.0 controller drives two SATA 3.0 ports for hard drives:

- The SATA 3.0 is a Marvell 88SE9170 SATA 3.0 controller with a ×1 Gen 2 link to the PCIe switch.
- The connections between the SATA 3.0 ports and the SATA 3.0 controller have a Serial ATA Generation 3 transfer rate of 6Gbps.

3.7.4 Gigabit Ethernet port

The PCIe switch connects to the Gigabit Ethernet (GbE) controller over a ×1 Gen 1.1 link.

The GbE controller drives an RJ45 GbE port on the back panel.

- The GbE controller is a RealTek RTL8111GS device.
- The controller provides a 10/100/1000Base-T connection to the GbE port.

3.7.5 USB 3.0 ports

The PCI switch connects to a USB 3.0 Host controller over a single-lane PCIe Gen 2 link.

The USB 3.0 Host controller drives four USB 3.0 ports on the back panel:

- The USB 3.0 controller is a Texas Instruments TUSB7340IRKMT device with a Gen 2 link to the PCIe switch.
- The controller operates at USB 3.0, 5Gbps, and is backwards compatible with USB 2.0, 480Mbps.

Related information

[The Morello SDP at a glance](#) on page 19

3.8 UARTs

The Morello SDP UART system enables access to the Morello SoC and IOFPGA on the Morello board.

Overview of UART system

The Morello SDP UART system enables access to the Morello SoC.

Morello SoC provides:

- AP UART0 - Non-secure Application Processor UART.
- AP UART2 - Secure Application Processor UART.
- SCP UART - System Control Processor UART.
- MCP UART - Manageability Control Processor UART.

The features of the AP UART2 are:

- Secure UART for application processors.
- PL011 UART.
- APB3 interface.
- Transmit and receive FIFOs with 32 entries.
- 8-bit words.

The following figure shows the Morello SDP UART system.

The diagram illustrates the internal architecture of the Morello SoC and its external connections. The SoC is divided into three main functional blocks: Morello SoC, IOFPGA, and Serial\rightarrowUSB.

Morello SoC: Contains several processing units and their associated UARTs:

- AP (Application Processor) with UART1
- MCP (Media Control Processor) with UART1
- SCP (Security Coprocessor) with UART
- AP (Application Processor) with UART2

IOFPGA: Contains programmable logic elements:

- UART IP0 and UART IP1

Serial\rightarrowUSB: Contains multiple ports for serial-to-USB conversion:

- Port 1 MCC
- Port 2 PCC
- Port 3 AP0
- Port 4 SCP
- Port 5 MCP
- Port 6 IPO
- Port 7 IP1
- Port 8 AP2

External Components and Connections:

- MCC (Media Control Core):** Connected to Port 1 MCC via UART1 and has a USB interface.
- PCC (Processing Core):** Connected to Port 2 PCC via UART1, Port 3 AP0 via UART5, Port 4 SCP via UART4, and Port 5 MCP via UART6. It also has a USB interface.
- USB Hub:** Receives data from the USB interfaces of the MCC, PCC, and the Serial\rightarrowUSB block. It has multiple USB ports (USB1 to USB6) and a USB Type-B connector.
- ULINKplus NXP Microcontroller:** Connected to the USB Hub via USB - ULINK.
- Supervisor:** Connected to the Serial\rightarrowUSB block via F2USB_nRST.
- Config Eeprom:** Connected to the Serial\rightarrowUSB block.
- USB Type-B:** The external USB connector.

The system is configurable using the settings in the `config.txt` file. The following table shows the default UART connectivity settings.

Board component UART	Intermediate connection	Final connection	Comment
AP UART0	Serial<>USB bridge, PORT3.	DBG USB connector	-
AP2 UART	Serial<>USB bridge, PORT8.	DBG USB connector	-
SCP UART	Serial<>USB bridge, PORT4.	DBG USB connector	-
MCP UART	Serial<>USB bridge, PORT5.	DBG USB connector	-

Board component UART	Intermediate connection	Final connection	Comment
MCC UART	Serial<>USB bridge, PORT1.	DBG USB connector	This COM port is visible when a serial terminal display is connected to the DBG USB connector. The command prompt is not visible without mains power to the system.
PCC UART	Serial<>USB bridge, PORT2.	DBG USB connector	This COM port is visible when a serial terminal display is connected to the DBG USB port.
UART IPO	Serial<>USB bridge, PORT6.	DBG USB connector	-
UART IP1	Serial<>USB bridge, PORT7.	DBG USB connector	-



To reduce the number of serial ports advertised, a jumper can be fitted to disconnect serial ports 5-8. For more information, see [The Morello SDP at a glance](#) and [Getting started](#).

AP UART1 is used to communicate through MCP UART1 internally within the SoC and is accessible to the Application Processor (AP) cores. MCP UART1 is not accessible to the AP cores.

Operation and getting started

When a serial terminal is connected to the DBG USB port, but before mains power is applied:

- The USB powers the USB hub and the Serial<>USB bridge.
- The MCC and PCC UARTS are visible as COM ports.
- The command prompt is not shown.

When mains power is applied, the MCC and PCC are powered and the MCC command prompt is shown. The system is in the standby state waiting for a press of the PBON button to complete the powerup and configuration process. The UART system is configured according to the settings in the `config.txt` file.

See the following sections for more information:

- [Getting started](#)
- [config.txt board configuration file](#)

Mapping between variables in the `config.txt` file and the UART ports

The following table shows the mapping between the variables in the `config.txt` file and the UART connectors and UART ports:

Table 3-10: config.txt file variables

config.txt variable	UART interface or connector	Comment
USBPORT1	Serial<>USB bridge PORT1	Serial<>USB bridge connects to DBG USB connector through the USB hub.
USBPORT2	Serial<>USB bridge PORT2	-
USBPORT3	Serial<>USB bridge PORT3	-
USBPORT4	Serial<>USB bridge PORT4	-

config.txt variable	UART interface or connector	Comment
USBPORT5	Serial<>USB bridge PORT5	-
USBPORT6	Serial<>USB bridge PORT6	-
USBPORT7	Serial<>USB bridge PORT7	-
USBPORT8	Serial<>USB bridge PORT8	-
PCC_UART4	UART4 port on PCC	-
PCC_UART5	UART5 port on PCC	-
PCC_UART6	UART6 port on PCC	-

UART memory addresses

See [UART memory addresses and control registers](#) for information on the UART base memory addresses and control registers.

Related information

[The Morello SDP at a glance](#) on page 19

3.9 LEDs, switches, and buttons

There are system LEDs, user LEDs, user system buttons, configuration DIP switches and, user DIP switches on the Morello board.

3.9.1 MCC system LEDs

The following table shows the system LEDs associated with the *Motherboard Configuration Controller* (MCC).

Table 3-11: MCC system LEDs

LED	Description	Position	Access	Indicates
Power status	Three color RGY 4-level light pipe	Back panel second from bottom	-	ATX powered or faulty
System LEDs	Green 4-level light pipe	Back panel third from bottom	-	MCC USB activity
System LEDs	Orange 4-level light pipe	Back panel fourth from bottom	-	MCC USB activity

Related information

[The Morello SDP at a glance](#) on page 19

3.9.2 PCC system LEDs

The following table shows the system LEDs associated with the *Platform Controller Chip* (PCC).

Table 3-12: PCC system LEDs

LED	Description	Position	Access	Function
RAS RDIMM LEDs	2×orange	On Morello board. One next to each RDIMM slot	Remove side panel.	Indicates DDR4 configuration issues or that RAS events have occurred.
PCIe status LED	1×green	On Morello board. Near PCC.	-	Indicates that PCIe is properly configured and link training completed.
CCIX status LED	1×green	On Morello board. Near PCC.	-	Indicates that CCIX is properly configured and virtual channel setup completed.
UID LED	Blue 4-level light pipe	Back panel, first from bottom.	-	Unit Identification Light. Used to physically identify the system in a rack environment.
Numerical LED displays	2x green 7-segment displays	On Morello board, near board edge.	Remove side panel. Not visible through rear I/O opening.	Used by <i>System Control Processor</i> (SCP) and MCC to indicate system status through boot. Also used to indicate error conditions during normal run-time operations.

Related information

[The Morello SDP at a glance](#) on page 19

3.9.3 Miscellaneous LEDs

There are other system LEDs to indicate traffic to and from the Morello board.

The following table shows miscellaneous system LEDs.

Table 3-13: System LEDs

LED	Description	Position	Access	Indicates
Gigabit Ethernet LEDs	1×Yellow	GbE port on back panel	-	GbE activity
Gigabit Ethernet LEDs	1×Green-Orange	GbE port on back panel	-	GbE activity Green: 10/100 Link. Orange: 1000 Link.
SATA LEDs	2×green	On Morello board. One next to each SATA connector.	Remove side panel.	SATA activity.
USB 3.0 VBUS LEDs	4×green	On back panel near USB 3.0 connectors	-	USB VBUS

LED	Description	Position	Access	Indicates
PCC 10/100 Ethernet LEDs	1×yellow	PCC GbE port on back panel	-	PCC GbE activity
PCC 10/100 Ethernet LEDs	1×green	PCC GbE port on back panel	-	PCC GbE 10/100 Link

Related information

[The Morello SDP at a glance](#) on page 19

3.9.4 Push buttons and switches

There are reset push buttons, configuration DIP switches, and user DIP switches on the Morello board.

Reset push buttons

The hardware reset push buttons are:

- PBON, the ON/OFF button.
- PBRESET, the Hardware reset button.

See [Resets](#) for a description of the functions of the hardware reset push buttons. See [The Morello SDP at a glance](#) for the locations of the hardware reset buttons.

Configuration DIP switches

There are two configuration DIP switches, SW0 and SW1 on the back panel.

The following table shows the functions of the configuration DIP switches.

Table 3-14: Configuration DIP switches

Switch	OFF (Default)	ON
SW0	Reserved	Reserved
SW1	Disabled	Enable MCC hard reset from UART0, DSR .

See [Remote UART configuration](#) for information on enabling MCC hard reset from UART0, **DSR**.

IOFPGA DIP switches

There are eight DIP switches connected to the IOFPGA.

The IOFPGA DIP switches are reserved for future use.

The IOFPGA also drives eight green LEDs, also reserved for future use.

Related information

[The Morello SDP at a glance](#) on page 19

3.10 Debug

The Morello SDP provides on-chip CoreSight™ debug technology to enable P-JTAG and 32-bit trace debug.

The 20-pin box header on the back panel provides access to JTAG debug.

The trace connector on the back panel provides access to JTAG debug and to 32-bit trace.

See [The Morello SDP at a glance](#) for the location of the JTAG and trace connectors on the back panel. For more information on CoreSight™ debug technology, see *Arm® CoreSight™ Components Technical Reference Manual*.

Related information

[The Morello SDP at a glance](#) on page 19

3.11 Embedded ULINKplus

The Morello SDP contains an embedded debug and trace probe, ULINKplus™. This debug circuitry is a simplified implementation of the Arm® Keil® ULINKplus™ probe, revision 1.2. It implements the CMSIS-DAP interface to send and receive commands.

For more information, see *Arm® Keil® ULINK® User's Guide* and *Arm® Development Studio Morello Edition Getting Started Guide*.

3.11.1 ULINK mux control

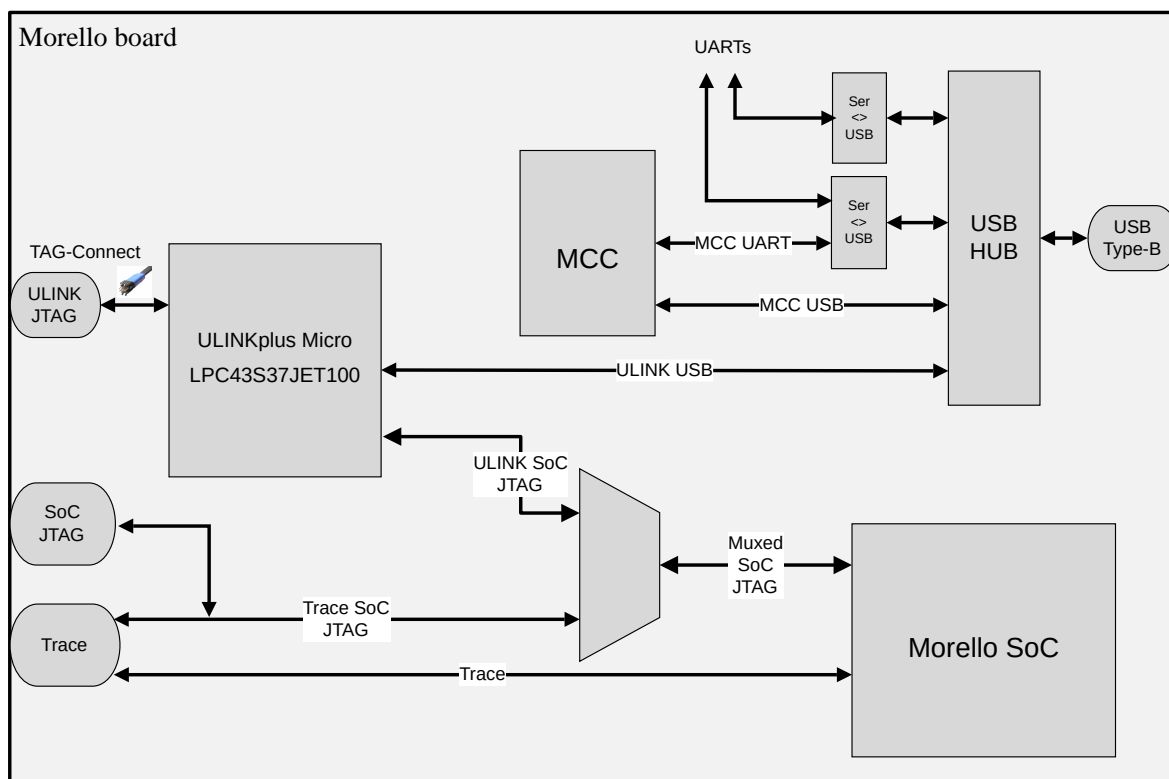
Multiplexed connectivity options are selected by the *Motherboard Configuration Controller* (MCC), and gating logic within the IOFPGA. The MCC drives the mux based on the value of the flags on the `config.txt` file, stored on the microSD card, and the detection of an external debugger connection.

The values control the mux as follows:

- `FALSE` The Embedded Arm® Keil® ULINKplus™ debugger is enabled unless an external Arm® DSTREAM debugger cable is detected.
- `TRUE` The Embedded ULINKplus debugger is enabled regardless of an external DSTREAM debugger connection.

The following figure shows the Embedded ULINKplus multiplexed options.

Figure 3-9: Embedded ULINKplus



Related information

[Overview of configuration files and microSD card directory structure on page 60](#)

4 Configuration

This chapter describes the powerup and configuration process of the Morello SDP.

4.1 Overview of the configuration process

The *Motherboard Configuration Controller* (MCC), the *Platform Controller Chip* (PCC) together with the configuration microSD card, configure the Morello SDP during powerup or reset.

The MCC uses information in the configuration EEPROM and the configuration microSD card during the configuration process, including the board HBI number.



Note

- The HBI number is a unique code that identifies the board. The root directories of the microSD card contain subdirectories in the form HBI<BoardNumber><BoardRevision>, for example HBI0316A. HBI0364A is the HBI number of the Morello board.
- If the MCC does not find a configuration directory that matches the HBI number of the board, the configuration process fails and the board reenters the standby state.

The configuration microSD card stores Morello SDP configuration files, including the `board.txt` file.

When power is applied to the Morello board by pressing the power button on the PC case, or the PBON button, preliminary configuration takes place and the board enters the standby state.

The MCC command-line interface is enabled in the standby state. You can connect a workstation to the DBG USB port and edit configuration files or Drag and Drop new configuration files.

Configuration resumes after another press of the power button on the PC tower, or the PBON button. The configuration process then completes without further intervention from the user. The system is then in operating state and application code runs.

A long press of the PBON button, longer than two seconds, initiates a software reset of the system and puts it into the standby state. Pressing the Hardware Reset button, PBRESET, initiates a forced reset and puts the system into standby state.

See [Getting started](#).

Related information

- [The Morello SDP at a glance](#)
- [Getting started](#)

4.2 Powerup and powerdown sequences

The ON/OFF button, PBON, the Hardware Reset button, PBRESET, and powerdown requests from the operating system initiate the powerup and powerdown sequences.

Powerup sequence from powered down state

The powerup sequence of the Morello board is as follows:

1. The PC tower is switched on using the power switch.
2. The *Motherboard Configuration Controller* (MCC) and the *Platform Controller Chip* (PCC) are powered from SB_3V3. All other supplies are powered off.
3. The MCC powers the EEPROM on the board and reads it to determine the HBI code for the board.
4. The system enables the MCC command-line interface on UART0.
5. The system enables the configuration microSD card. You can connect a workstation to the DBG USB port to edit existing configuration files or Drag-and-Drop new configuration files.
6. The system waits in standby state.
7. The PBON button is pressed briefly.
8. The system loads the board configuration file:
 - The MCC reads the generic `config.txt` file.
 - The MCC searches the configuration microSD card MB directory for a directory name that matches the board HBI number.
9. If the MCC finds configuration subdirectories that match the HBI code of the board, configuration continues and the MCC reads the `board.txt` file.
10. If the MCC does not find the correct configuration subdirectories or files, it records the failure to a log file on configuration microSD card. Configuration stops and the system reenters the standby state.
11. The MCC enables the ATXPSU (ATXON).
12. The MCC takes the PCC out of reset in the Enterprise user case.
13. The MCC enables all the supplies, including the board VIO, and the Morello SoC and IOFPGA supplies.
14. The MCC enables the *System Control Processor* (SCP) 32kHz clock, **REFCLK**, and the board clocks.
15. The MCC reads the FPGA image from the configuration microSD card and loads it into the IOFPGA.
16. The MCC sets the board oscillator frequencies using values from the `board.txt` file.
17. The MCC releases the *Serial Configuration Controller* (SCC) reset, **nCFG_RESET**.
18. If necessary, the MCC programs the IOFPGA and Morello SoC SCC registers as a backup procedure.
19. The MCC programs the SCP and MCP QSPI images from the `images.txt` file.

20. The MCC notifies the PCC that the Morello SoC is coming out of reset and then releases **nSRST**.
21. The MCC releases **nPOR** to the Morello SoC.
22. The MCC enters run state enabling the USBMSD and monitors the MCC UART interface for user commands. It also communicates with the PCC.
23. The SCP and *Manageability Control Processor* (MCP) boot from internal ROM.
24. The SCP and MCP run code from their QSPI flash.
25. The SCP performs the basic Morello SoC setup, PLLs, internal clocks.
26. The SCP releases the *Power Policy Units* (PPUs) to begin the application boot sequence.
27. Application code runs on the Morello SoC. The system is in the operating state.

Powerdown sequence

A long press, greater than two seconds, of the PBON button, initiates the powerdown sequence. The powerdown sequence is as follows:

1. Press the PBON button for longer than two seconds.
2. The PCC detects the power request and sequences the power down with the SCP.
3. The SCP signals the powerdown request to the application processor, that is, one of the Morello clusters.
4. The application cluster goes through its cleanup and shutdown sequence.
5. The application cluster goes to the *Wait for Interrupt* (WFI) state.
6. The PPU sees the WFI state and powers down. The SCP waits for this sequence to complete.
7. The SCP powers down all supplies.
8. The SCP signals the PCC that it is ready for shutdown using the I²C bus.
9. The PCC requests a powerdown from the MCC using the SPI bus.
10. The MCC asserts **nPOR**, disables the board clocks and the ATXPSU.
11. The system is now in the standby state and waits for a short press of the PBON button.

Related information

[The Morello SDP at a glance](#) on page 19

4.3 Configuration files

Configuration files in the configuration microSD card control the board powerup and configuration process.

4.3.1 Overview of configuration files and microSD card directory structure

Because the Morello SDP configuration microSD card is non-volatile flash memory, it is only necessary to load new configuration files if you change the system configuration. The configuration microSD card contains default configuration files.

If you connect a workstation to the DBG USB port, the microSD card appears as a *USB Mass Storage Device* (USBMSD) and you can add, edit, or delete files.

You can use a standard text editor that produces DOS line endings to read and edit the board configuration files.

The following figure shows a typical example of the directory structure in the microSD card memory.

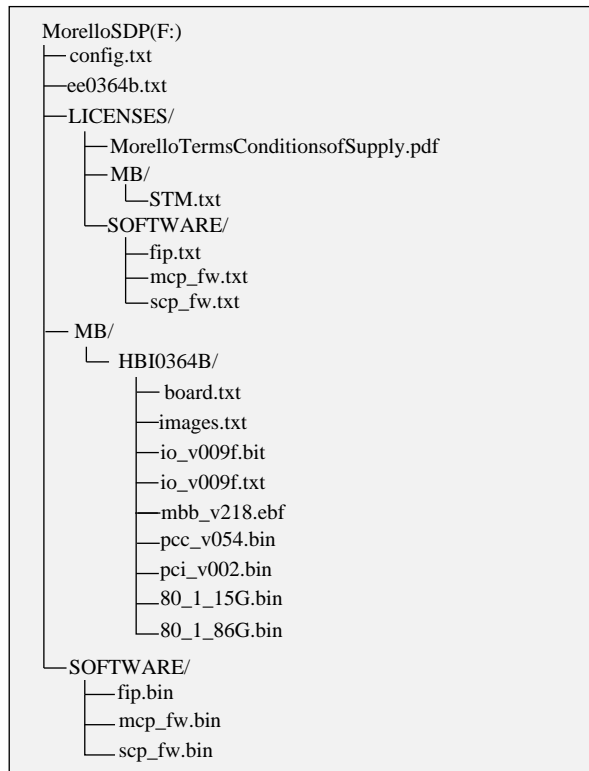


Caution

Files names and directory names are in 8.3 format:

- File names that you generate must be in lowercase.
- Directory names must be in uppercase.
- All configuration files must end in DOS line endings, 0x0D/0x0A.

Figure 4-1: Example configuration microSD card directory structure



The directory structure and file name format ensure that each image is matched to the correct target device defined in the Morello SDP configuration EEPROM.

config.txt file

Generic configuration file for all motherboards. This file applies to all Arm® development boards including the Morello board.

LICENSES directory

Contains license text files.

MB directory

Contains one or more HBI subdirectories for board variants. The subdirectory names match the HBI codes for the board variants.

HBI0364B? subdirectories

Contain pointer files that define BIOS images for the MCC, the Power Management IC (PMIC), and the IOFPGA, SCP, and MCP. Other files contain image, and configuration information for the board and IOFPGA.

SOFTWARE directory

Contains application files for the SCP and MCP.

4.3.2 config.txt board configuration file

You can connect a workstation to the DBG USB port to update the generic Arm development board configuration file `config.txt` in the root directory of the microSD card.

The following example shows a typical `config.txt` configuration file in the root directory of the configuration microSD card.



- Colons (:) indicate the end of commands and must be separated by a space character (0x20) from the value fields.
- Semicolons (;) indicate comments.

```
BOARD: HBI0364B
TITLE: M1SDP Configuration File

[CONFIGURATION]
TESTMENU: FALSE
AUTORUN: TRUE           ;Auto Run from power on
AUTORUNDELAY: 3         ;Delay in seconds to wait for key press to stop boot up

RTC: TRUE               ;TRUE = Enable RTC, False = Disable RTC

APUART0: 0              ;0-USBPORT_3 1-PCCUART_5
SCPUART: 0              ;0-USBPORT_4 1-PCCUART_4
MCPUART0: 0            ;0-USBPORT_5 1-PCCUART_6

EMBEDDED_ULINK_DEBUG: FALSE ;FALSE - Embedded UlinkPlus debugger enabled unless and
                           external DStream debugger cable is detected
                           ;TRUE - Embedded UlinkPlus debugger enabled regardless
                           of external DStream debugger connection
```

```
VOLTAGEGUARD: TRUE           ;TRUE - Voltage error results in Board power down
                             ;FALSE - Under/Over Voltage error, user can either
                             proceed or halt boot
DVIMODE: VGA                 ;VGA/SVGA/XGA/SXGA/UXGA or HD1080 (MCC sets OSCCLK5)
CONFSWITCH: 01010101        ;Configuration Switch[7:0] in binary
CASE_FAN_SPEED: START        ;START (default speed) /SLOW/MEDIUM/FAST/10%/20%/.../100%
USB_REMOTE: FALSE            ;Selects remote command via USB !TBA
```

See [UARTs](#) for information on configuring the UARTs and UART connectivity.

4.3.3 Contents of the MB directory

The MB directory contains a configuration HBIxxxx subdirectory that matches the HBI code of the Morello board. The HBIxxxx subdirectory contains files that relate to the MCC and to other components on the Morello board, but not the Morello SoC.

The HBI subdirectory contains the following:

One `board.txt` file

Points to the BIOS image that the MCC loads during configuration.

One `images.txt` file

Points to the SCP and MCP image files.

Files of the form `io_vxxx.bit`

IOFPGA image files for different board variants.

File of the form `io_vxxx.txt`

IOFPGA configuration files for different board variants.

A file of the form `mbb_vxxx.ebf`.

MCC BIOS image that the `board.txt` file defines.

A file of the form `pms_vxxx.bin`

BIOS image for the *Power Management IC* (PMIC) on the Morello board.

The following example shows a typical Morello board configuration `board.txt` file.

```
BOARD: HBI0364B
TITLE: M1SDP Motherboard Configuration File

[MCCS]
MBBIOS: mbb_v218.ebf           ;MB BIOS IMAGE

[PCCS]
PCCBIOS: pcc_v054.bin          ;PCC BIOS IMAGE

[PCIE]
MBPCIE: pci_v002.bin           ;MB PCIE SWITCH CONFIG

[FPGA]
APPFILE: io_v009f.txt          ;Please select the required IOFPGA image
                                ;M1SDP IOFPGA image Morello
```

The following example shows a typical Morello board `images.txt` file.

```
BOARD: HBI0364B
TITLE: M1SDP Images Configuration File

[IMAGES]
TOTALIMAGES: 3 ;Number of Images (Max: 32)

IMAGE0ADDRESS: 0x64000000 ;Please select the required executable program
IMAGE0UPDATE: MCP_AUTO ;Image Update:NONE/AUTO/FORCE/SCP_AUTO/MCP_AUTO/AP_AUTO
IMAGE0FILE: \SOFTWARE\mcp_fw.bin ;Image for test

IMAGE1ADDRESS: 0x64000000 ;Please select the required executable program
IMAGE1UPDATE: SCP_AUTO ;Image Update:NONE/AUTO/FORCE/SCP_AUTO/MCP_AUTO/AP_AUTO
IMAGE1FILE: \SOFTWARE\scp_fw.bin ;Image for test

IMAGE2ADDRESS: 0x66000000 ;Please select the required executable program
IMAGE2UPDATE: AP_AUTO ;Image Update:NONE/AUTO/FORCE/SCP_AUTO/MCP_AUTO/AP_AUTO
IMAGE2FILE: \SOFTWARE\fip.bin ;Image for test
```

The following example shows a typical Morello board configuration file, `io_v009f.txt`.

```
BOARD: HBI0364B
TITLE: M1SDP IOFPGA Configuration File

[FPGAS]
TOTALFPGAS: 1 ;Total Number of FPGAs
F0FILE: io_v009f.bit ;FPGA0 Filename
F0MODE: FPG_A ;FPGA0 Programming Mode

[PMIC]
MBPMIC: 80_1_86G.bin ;MB PMIC

[OSCCCLKS]
TOTALOSCCCLKS: 15
OSC0: 50.0 ;OSC0-Y5 - SYS_REF_CLK
OSC1: 50.0 ;OSC0-Y4 - CPU0_REF_CLK
OSC2: 50.0 ;OSC0-Y6 - CPU1_REF_CLK
OSC3: 50.0 ;OSC1-Y2 - CLUS_REF_CLK
OSC4: 50.0 ;OSC1-Y4 - INT_REF_CLK
OSC5: 50.0 ;OSC1-Y6 - REF_CLK
OSC6: 50.0 ;OSC2-Y2 - DMC_REF_CLK
OSC7: 50.0 ;OSC2-Y4 - PXL_REF_CLK
OSC8: 75.0 ;OSC2-Y6 - IOFPGA_TLXCLK (Not Used - Reserved)
OSC9: 85.0 ;OSC3-Y2 - IOFPGA_ACLK
OSC10: 12.288 ;OSC3-Y4 - IOFPGA_AUDCLK
OSC11: 50.0 ;OSC3-Y6 - IOFPGA_SPARECLK
OSC12: 50.0 ;OSC3-Y7 - DPU_REF_CLK
OSC13: 50.0 ;OSC1-Y5 - GPU_REF_CLK
OSC14: 170.0 ;OSC2-Y7 - IOFPGA_TLX2XCLK

[PERIPHERAL SUPPORT]
FPGA_SMB: TRUE ;SMB interface is supported (MCC_SMC<>FPGA_SMB)

FPGA_SCC: TRUE ;SCC interface is supported
SCCREG: 0x68130000 ;SCC registers base address

FPGA_DDR: TRUE ;DDR interface is supported
DDRBASE: 0x68040000 ;DDR I2C register address

FPGA_SYSREG: TRUE ;System register interface is supported
FPGAREG: 0x68010000 ;System registers base address
```

```
FPGA_HDMI: TRUE           ;HDMI interface is supported
HDMIBASE: 0x680F0000      ;HDMI I2C register address

FPGA_LAN: TRUE            ;LAN LAN9220 interface is supported
LANBASE: 0x69100000       ;LAN LAN9220 base address

FPGA_RTC: TRUE            ;RTC PL031 interface is supported
RTCBASE: 0x68100000       ;RTC PL031 base address

FPGA_QSPI: TRUE           ;QSPI interface is supported
QSPIBASE: 0x680C0000      ;QSPI controller base address
QSPIDATA: 0x64000000      ;QSPI data address

[SCC REGISTERS]
TOTALSYSCONS: 15          ;Total Number of SCC registers defined
SYSCON: 0x0020 0x04030201 ;IOFPGA_CFG_REG8 Bits[31:24] SOC_IRQ[5] MUXsel,
[23:16] SOC_IRQ[4] MUXsel, [15:8] SOC_IRQ[3] MUXsel, [7:0] AP_EXT_INT MUXsel
SYSCON: 0x0024 0x08070605 ;IOFPGA_CFG_REG9 Bits[31:24] MCP_EXT_INT MUXsel,
[23:16] SCP_EXT_INT MUXsel, [15:8] SOC_IRQ[7] MUXsel, [7:0] SOC_IRQ[6] MUXsel
SYSCON: 0x0028 0x00000A09 ;IOFPGA_CFG_REG10 Bits[15:8] F2PCC_INT MUXsel,
[7:0] F2MCC_INT MUXsel

SOCCON: 0x1160 0x00000001 ;SoC SCC BOOT_CTL - enable TLX
SOCCON: 0x1164 0x01000000 ;SoC SCC BOOT_CTL_STA (0xX1000000 = MCC OK)
SOCCON: 0x1168 0x00000000 ;SoC SCC SCP_BOOT_ADR
SOCCON: 0x116C 0x00000000 ;SoC SCC MCP_BOOT_ADR

; BOOT_GPR0 - SCP RAMFW Capsule base
; Value of 0x0 means SCP ROM looks for capsule at 0x30000000
; Non-zero value means SCP ROM looks for capsule at programmed base address
SOCCON: 0x1180 0x00000000 ;SoC SCC BOOT_GPR0

; BOOT_GPR1 - HW/FW configuration register
; BIT0 - Bing mode (0 - Server mode, 1 - Client mode)
; BIT1 - Reserved
; BIT2 - Bing C1 tag cache control (0 - Disable, 1 - Enable)
; BIT3 - Bing C2 tag cache control (0 - Disable, 1 - Enable)
; BIT4 - CPU Cluster L3 cache control (0 - Disable, 1 - Enable)
; BIT31:5 - Reserved
SOCCON: 0x1184 0x0000000C ;SoC SCC BOOT_GPR1

; BOOT_GPR2, BOOT_GPR3 - Application Processor RVBAR base address
; BOOT_GPR3 - RVBAR_HIGH, BOOT_GPR2 - RVBAR_LOW
SOCCON: 0x1188 0x00000000 ;SoC SCC BOOT_GPR2
SOCCON: 0x118C 0x00000000 ;SoC SCC BOOT_GPR3

SOCCON: 0x1190 0x00000000 ;SoC SCC BOOT_GPR4
SOCCON: 0x1194 0x00000000 ;SoC SCC BOOT_GPR5
SOCCON: 0x1198 0x00000000 ;SoC SCC BOOT_GPR6
SOCCON: 0x119C 0x00000000 ;SoC SCC BOOT_GPR7
```

4.3.4 Contents of the SOFTWARE subdirectory

The `SOFTWARE` subdirectory contains applications that you can load into external flash memory.

You can create new applications and load them into the flash memory on the Morello SDP. Application images are typically boot images or demo programs. The system recognizes `.elf`, `.bin`, and `.bit` files.

Typical applications in this subdirectory are:

- `mcp_fw.elf` MCP image file.
- `scp_fw.elf` SCP image file.

4.4 Configuration switches

There are configuration switches SW0 and SW1 on the back panel.

4.4.1 Use of configuration switches

The SW0 and SW1 switches affect board initialization.

The `config.txt` configuration file contains `USERSWITCH` and `CONF SWITCH` entries for the virtual switch register bits `SYS_SW[7:0]` and `SYS_CFGSW[7:0]` in the IOFPGA. The configuration system does not use these virtual switches for system configuration, but they are available for the user application and boot monitor.



- The default setting for configuration switches SW0 and SW1 is OFF.
 - If the switches are in the up position, they are OFF. See [Getting started](#).
-

Boot script switch SW0

Reserved.

Remote UART control switch SW1

SW1 in the ON position enables UART control and the flow-control signals on UART0 to control the standby-state. This setting is typically used on test farms.

See the following for information about the configuration switches:

- [The Morello SDP at a glance](#)
- [Getting started](#)

4.4.2 Remote UART configuration

To enable remote UART control:

Switch SW1, on the side panel must be ON, and the correct options must be set in the `config.txt` file to connect the UART0 port to the MCC.

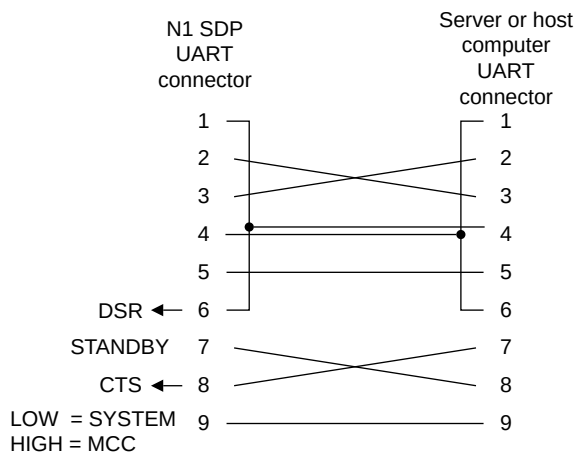
An external controller can toggle the UART0 connector on the side panel **SER0_DSR**, pin 6, HIGH for 100ms to put the Morello SDP into standby-state. This is equivalent to pressing the Hardware Reset button, PBRESET. Power cycling the board also places the system in the standby state.



The duration of the **SER0_DSR** HIGH pulse must be greater than or equal to 100ms.

Remote UART0 control requires a full null modem cable that Arm supplies with the Morello SDP. The following figure shows the cable wiring.

Figure 4-2: Modem cable wiring



You can control the **SER0_DSR** and **SER0_CTS** signals using control logic on the host computer.

Alternatively, you can use a custom terminal program such as `vtTerminal.exe` that Arm provides on the Morello SDP DVD. This program integrates the terminal output and control buttons into a single application.

Related information

[The Morello SDP at a glance](#)

4.5 Use of reset push buttons

The ON/OFF button, PBON, initiates the powerup and powerdown sequences. The Hardware reset button, PBRESET, initiates a hardware reset.

In standby state, press the PBON button briefly, less than two seconds, to initiate the powerup sequence.

In the operating state, press the PBON button for longer than two seconds to initiate the powerdown sequence.

In the operating state, press the PBRESET button to initiate a hardware reset.

See [Powerup and powerdown sequences](#).

Related information

[The Morello SDP at a glance](#) on page 19

4.6 Command-line interface

The Morello SDP command-line interface supports system command-line input to the *Motherboard Configuration Controller* (MCC).

4.6.1 Overview of the Morello SDP MCC command-line interface

You must connect a workstation to the DBG USB to enter MCC system commands at the MCC USB port.

You must set the `MBLOG` option in the `config.txt` to `TRUE` to enter MCC system commands.

The workstation settings must be:

- 115.2kBaud.
- Morello representing 8 data bits, no parity, one stop bit.
- No hardware or software flow control.

Related information

[The Morello SDP at a glance](#)

4.6.2 MCC main command menu

The following table shows the MCC main menu system commands.

Table 4-1: Morello SDP MCC main command menu

Command	Description
CAP "fname" [/A]	Captures serial data to the file. Use the /A option to append data to an existing file.
FILL "fname" [nnnn]	Create a file filled with text. nnnn specifies the number of lines to create. The default is 1000.
TYPE "fname"	Display the contents of text file.
REN "fname1" "fname2"	Rename a file from fname1 to fname2.

Command	Description
COPY "fin" ["fin2"] "fout"	Copy a file <code>fin</code> to <code>fout</code> . [<code>fin2</code> option merges <code>fin</code> and <code>fin2</code>].
DEL "fname"	Deletes file.
DIR "[mask]"	Display a list of files in the directory.
FORMAT [label]	Formats Flash memory card.
USB_INIT	Re-initialize USB.
USB_ON	Enable MCC USB configuration port.
USB_OFF	Disable MCC USB configuration port.
SHUTDOWN	Shut down the power supply but leave the MCC running. The board returns to Standby mode.
REBOOT	Cycle system power and reboot.
RESET	Reset the SoC using the SoC_nSRST reset signal.
IOFPGA_VERSION	Displays IOFPGA version.
DEBUG	Change to the debug menu.
EEPROM	Change to the EEPROM menu.
HELP or ?	Display this help.

The following table shows MCC main menu system commands that are only available in Run mode.

Table 4-2: Morello SDP MCC main command menu Run mode

Command	Description
CASE_FAN_SPEED "SPEED"	SLOW, MEDIUM, FAST, 0%, 10%, 20%, .. 100%
READ_AXI "fname" "address" "end_address"	Read system memory to file <code>fname</code> from address to end address.
WRITE_AXI "fname" "address"	Write a file <code>fname</code> to system memory at address.

4.6.3 MCC debug menu

To switch to the debug submenu enter `DEBUG` at the main menu. The debug submenu is valid only in operating state.

The following table shows the debug commands.

Table 4-3: Morello SDP MCC debug command menu

Command	Description
DATE	Displays current date.
TIME	Displays current time.
RTC	Displays current RTC (PL031) time.
TEMP	Displays current temperature.
VOLTS	Displays current voltage.
UART	Displays FPGA UART test.
SELFTEST	Runs the selftest menu.

Command	Description
CFG "R:W" "OSC :V :TEMP :SCC : " "device" ["data"]	Read/Write SPI configuration command. For SCC, device is the register address.
DCFG "W" "DVI" "VGA :SVGA :XGA :SXGA :VXGA"	Write DVI configuration command.
DEBUG "0:1"	Enable(1) or Disable(0) debug printing.
EXAM "address" [nnnn]	Examine system memory address, nnnn is number, in Hex, of words to read.
DEPOSIT "address" "data"	Write word to system memory address.
WRITE_SPI "register" "data"	Write word to SPI register in PCC.
READ_SPI "register"	Read SPI register from PCC.
SET_SPI "register"	Set SPI register in MCC to initial value.
EXAM_SPI "register"	Read SPI register in MCC.
DISPLAY_SPI	Displays voltage and temperature SPI registers.
READ_PMIC "channel"	Read PMIC channel return value in mV.
SET_PMIC "channel" "data"	Set PMIC channel with data in mV.
DISPLAY_PMIC	Displays voltage and current PMIC registers.
REBOOT	Power cycle system and reboot.
EXIT or QUIT	Returns to main menu.
HELP or ?	Displays this help.

4.6.4 EEPROM menu

To switch to the EEPROM submenu enter EEPROM at the main menu. The contents of the Morello SDP EEPROMs identify the specific board variant and might contain data to load to the other devices on the board.

The following table shows the EEPROM commands.



Caution

You must not modify the EEPROM settings. The settings are programmed with unique values during production and changing them might compromise the function of the board.

Table 4-4: Morello SDP EEPROM commands

Command	Description
CONFIG [0] "fname"	Writes configuration file to EEPROM.
READCF [0]	Read configuration EEPROM.
ERASECON [0]	Erase configuration section of EEPROM.
ERASEDEV [0]	Erase device section of EEPROM.
READRANGE [0] [start] [end]	Read EEPROM between start and end .
ERASERANGE [0] [start] [end]	Erase EEPROM between start and end .
READIMAGES	Read images stored in Motherboard EEPROM.

Command	Description
ERASEIMAGES	Erase images stored in Motherboard EEPROM.
ERASEIMAGE [image_id]	Erase image, named <code>image_id</code> , stored in Motherboard EEPROM.
EXIT or QUIT	Returns to main menu.
HELP or ?	Displays this help.

5 Programmers model

This chapter describes the programmers model of the Morello SDP.

5.1 About this programmers model

The following information applies to all registers in this programmers model:

- Do not attempt to access reserved or unused address locations. Attempting to access these locations can result in **UNPREDICTABLE** behavior.
- Unless otherwise stated in the accompanying text:
 - Do not modify undefined register bits.
 - Ignore undefined register bits on reads.
 - All register bits are reset to a logic 0 by a system or powerup reset.
 - All register summary tables in this chapter describe register access types as follows:

RW

Read/write.

RO

Read-only.

WO

Write-only.

5.2 Morello SDP memory maps

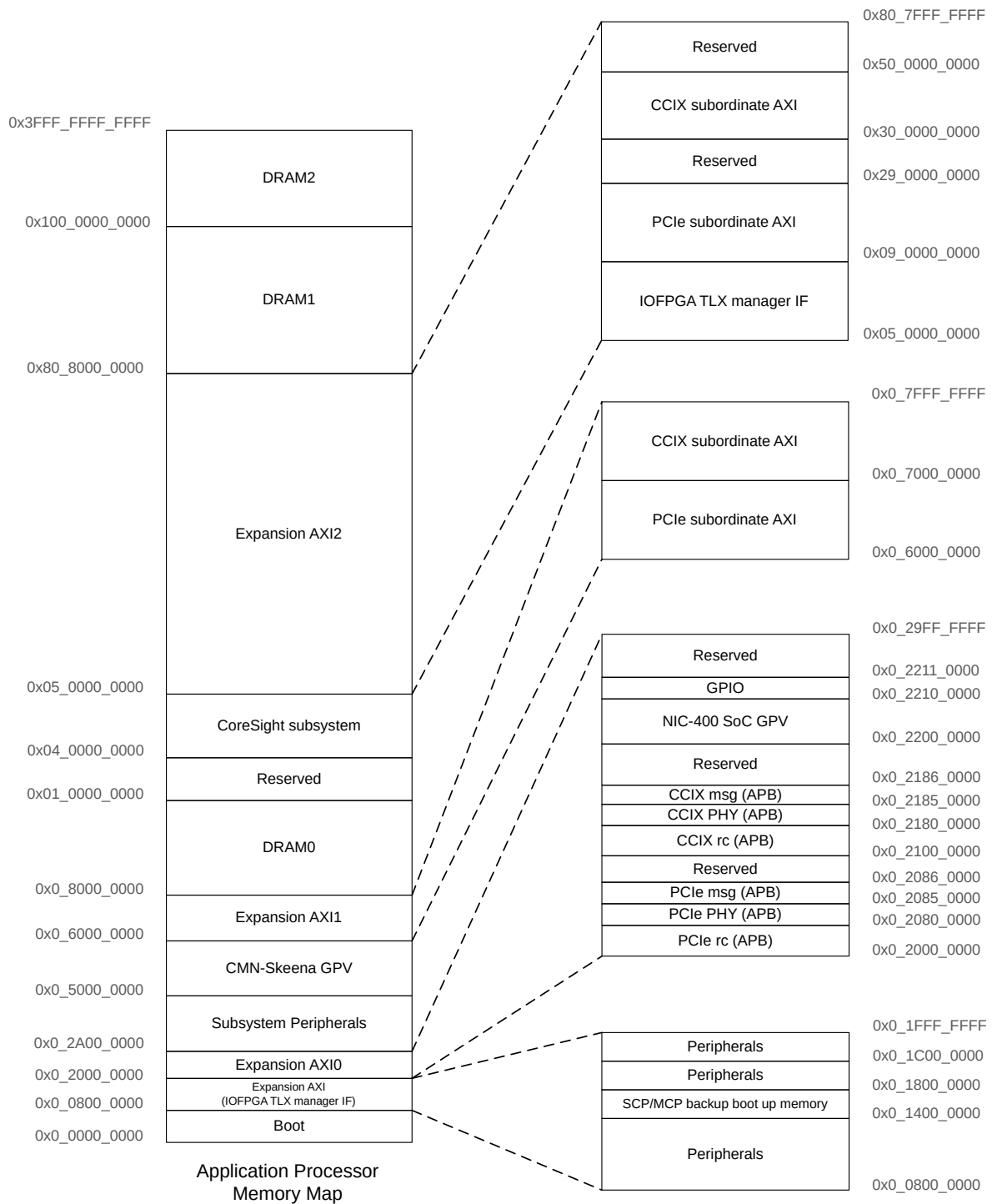
The Morello SDP contains Application Processor (AP), System Control Processor (SCP), and Manageability Control Processor (MCP) memory maps.

The SCP and MCP memory maps are private. The masters which can access the AP memory map have no access to the components in the SCP and MCP memory maps. Certain areas within the AP memory map are mapped into the SCP and MCP memory maps and the corresponding masters can access them.

5.2.1 Application Processor memory map

The following figure shows the Morello SDP Application Processor (AP) memory map.

Figure 5-1: AP memory map



The following table shows the Morello SDP AP memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-1: SDP AP memory map

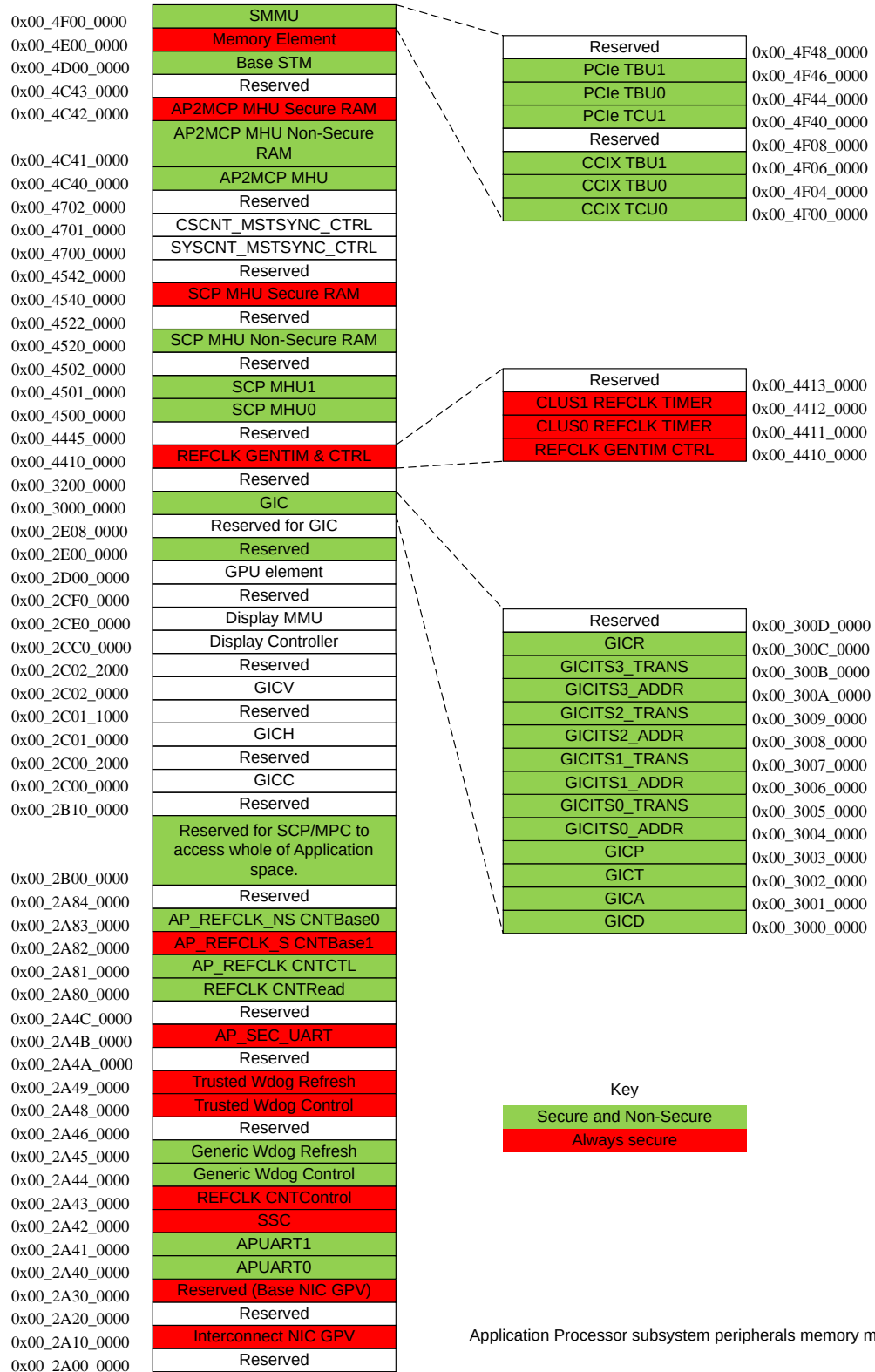
Address range from	Address range to	Size	Description
0x0_0000_0000	0x0_07FF_FFFF	128MB	Boot region
0x0_0800_0000	0x0_13FF_FFFF	192MB	Peripherals
0x0_1400_0000	0x0_17FF_FFFF	64MB	SCP, MCP backup boot memory
0x0_1800_0000	0x0_1BFF_FFFF	64MB	Peripherals
0x0_1C00_0000	0x0_1FFF_FFFF	64MB	Peripherals
0x0_2000_0000	0x0_207F_FFFF	8MB	PCIe rc (APB)
0x0_2080_0000	0x0_2084_FFFF	320KB	PCIe PHY (APB)
0x0_2085_0000	0x0_2085_FFFF	64KB	PCIe msg (APB)
0x0_2100_0000	0x0_217F_FFFF	8MB	CCIX rc (APB)
0x0_2180_0000	0x0_2184_FFFF	320KB	CCIX PHY (APB)
0x0_2185_0000	0x0_2185_FFFF	64KB	CCIX msg (APB)
0x0_2200_0000	0x0_220F_FFFF	1MB	NIC-400 SoC GPV
0x0_2210_0000	0x0_2210_FFFF	64KB	GPIO
0x0_2A00_0000	0x0_4FFF_FFFF	608MB	Subsystem peripherals
0x0_5000_0000	0x0_5FFF_FFFF	256MB	CMN-Skeena GPV
0x0_6000_0000	0x0_6FFF_FFFF	256MB	PCIe subordinate AXI
0x0_7000_0000	0x0_7FFF_FFFF	256MB	CCIX subordinate AXI
0x0_8000_0000	0x0_FFFF_FFFF	2GB	DRAM0
0x4_0000_0000	0x4_FFFF_FFFF	4GB	CoreSight subsystem
0x5_0000_0000	0x8_FFFF_FFFF	16GB	IOFPGA TLX manager IF
0x9_0000_0000	0x28_FFFF_FFFF	128GB	PCIe subordinate AXI
0x30_0000_0000	0x4F_FFFF_FFFF	128GB	CCIX subordinate AXI
0x80_8000_0000	0xFF_FFFF_FFFF	510GB	DRAM1
0x100_0000_0000	0x3FF_FFFF_FFFF	3TB	DRAM2

5.2.2 Application Processor subsystem peripherals memory map

The Application Processor (AP) memory map of the Morello SDP contains a region associated with the subsystem peripherals.

The following figure shows the subsystem peripherals region of the AP memory map.

Figure 5-2: Subsystem peripherals region of Application Processor memory map



The following table shows the subsystem peripherals region of the AP memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-2: Subsystem peripherals region of the AP memory map

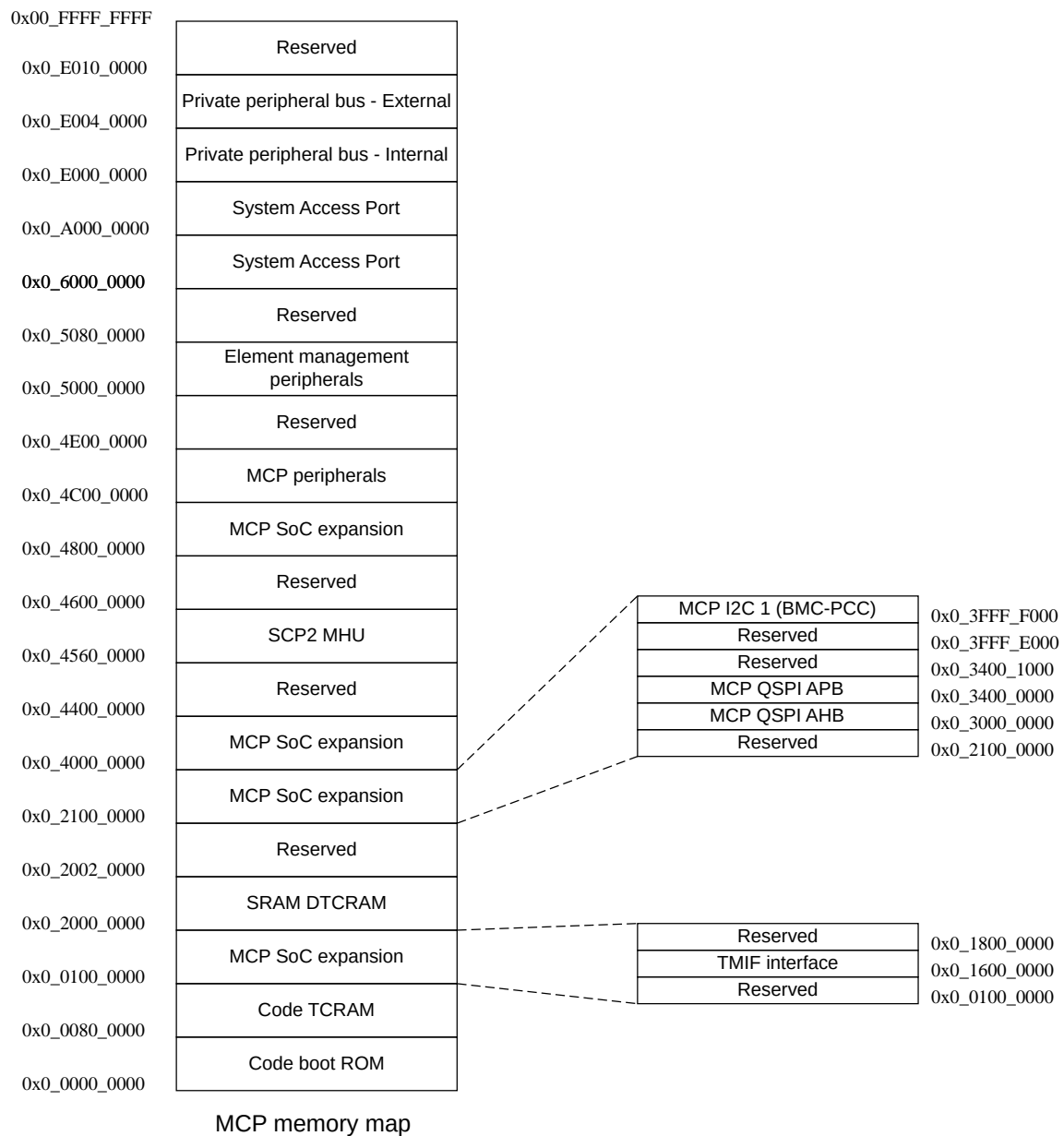
Address range from	Address range to	Size	Description
0x00_2A00_0000	0x00_2A0F_FFFF	1MB	Reserved
0x00_2A10_0000	0x00_2A1F_FFFF	1MB	Interconnect NIC GPV
0x00_2A30_0000	0x00_2A3F_FFFF	1MB	Reserved (Base NIC GPV)
0x00_2A40_0000	0x00_2A40_FFFF	64KB	APUART0
0x00_2A41_0000	0x00_2A41_FFFF	64KB	APUART1
0x00_2A42_0000	0x00_2A42_FFFF	64KB	System Security Control (SSC) registers
0x00_2A43_0000	0x00_2A43_FFFF	64KB	REFCLK CNTControl
0x00_2A44_0000	0x00_2A44_FFFF	64KB	Generic Watchdog Control
0x00_2A45_0000	0x00_2A45_FFFF	64KB	Generic Watchdog Refresh
0x00_2A48_0000	0x00_2A48_FFFF	64KB	Trusted Watchdog Control
0x00_2A49_0000	0x00_2A49_FFFF	64KB	Trusted Watchdog Refresh
0x00_2A4B_0000	0x00_2A4B_FFFF	64KB	AP_SEC_UART
0x00_2A80_0000	0x00_2A80_FFFF	64KB	REFCLK CNTRead
0x00_2A81_0000	0x00_2A81_FFFF	64KB	AP_REFCLK CNTCTL
0x00_2A82_0000	0x00_2A82_FFFF	64KB	AP_REFCLK_S CNTBase1
0x00_2A83_0000	0x00_2A83_FFFF	64KB	AP_REFCLK_NS CNTBase0
0x00_2B00_0000	0x00_2B0F_FFFF	1MB	Reserved for SCP/MCP to access whole of application space.
0x00_2C00_0000	0x00_2C00_1FFF	8KB	GICC registers
0x00_2C01_0000	0x00_2C01_0FFF	4KB	GICH registers
0x00_2C02_0000	0x00_2C02_1FFF	8KB	GICV registers
0x00_2CC0_0000	0x00_2CDF_FFFF	2MB	Display Controller
0x00_2CE0_0000	0x00_2CEF_FFFF	1MB	Display MMU
0x00_2D00_0000	0x00_2DFF_FFFF	16MB	GPU Element
0x00_2E08_0000	0x00_2FFF_FFFF	32MB	Reserved for GIC
0x00_3000_0000	0x00_3000_FFFF	64KB	GICD registers
0x00_3001_0000	0x00_3001_FFFF	64KB	GICA registers
0x00_3002_0000	0x00_3002_FFFF	64KB	GICT registers
0x00_3003_0000	0x00_3003_FFFF	64KB	GICP registers
0x00_3004_0000	0x00_3004_FFFF	64KB	GICITS0 ITS address
0x00_3005_0000	0x00_3005_FFFF	64KB	GICITS0 translater
0x00_3006_0000	0x00_3006_FFFF	64KB	GICITS1 address
0x00_3007_0000	0x00_3007_FFFF	64KB	GICITS1 translater
0x00_3008_0000	0x00_3008_FFFF	64KB	GICITS2 address
0x00_3009_0000	0x00_3009_FFFF	64KB	GICITS2 translater
0x00_300A_0000	0x00_300A_FFFF	64KB	GICITS3 address
0x00_300B_0000	0x00_300B_FFFF	64KB	GICITS3 translater
0x00_300C_0000	0x00_300C_FFFF	64KB	GICR registers

Address range from	Address range to	Size	Description
0x00_4410_0000	0x00_4410_FFFF	64KB	REFCLK general timer control
0x00_4411_0000	0x00_4411_FFFF	64KB	Cluster 0 time frame
0x00_4412_0000	0x00_4412_FFFF	64KB	Cluster 1 time frame
0x00_4500_0000	0x00_4500_FFFF	64KB	SCP <i>Message Handling Unit</i> (MHU) 0
0x00_4501_0000	0x00_4501_FFFF	64KB	SCP MHU1
0x00_4520_0000	0x00_4521_FFFF	128KB	SCP MHU Non-secure RAM
0x00_4540_0000	0x00_4541_FFFF	128KB	SCP MHU Secure RAM
0x00_4700_0000	0x00_4700_FFFF	64KB	SYSCNT_MSTSYN_CTRL
0x00_4701_0000	0x00_4701_FFFF	64KB	CSCNT_MSTSYNC_CTRL
0x00_4C40_0000	0x00_4C40_FFFF	64KB	AP2MCP MHU
0x00_4C41_0000	0x00_4C41_FFFF	64KB	AP2MCP MHU Non-Secure RAM
0x00_4C42_0000	0x00_4C42_FFFF	64KB	AP2MCP MHU Secure RAM
0x00_4D00_0000	0x00_4DFF_FFFF	16MB	Base STM
0x00_4E00_0000	0x00_4EFF_FFFF	16MB	Memory Element
0x00_4F00_0000	0x00_4F03_FFFF	256KB	<i>Translation Control Unit</i> (TCU) 0 for CCIX root port.
0x00_4F04_0000	0x00_4F05_FFFF	128KB	<i>Translation Buffer Unit</i> (TBU) 0 for CCIX root port.
0x00_4F06_0000	0x00_4F07_FFFF	128KB	<i>Translation Buffer Unit</i> (TBU) 1 for CCIX root port.
0x00_4F40_0000	0x00_4F43_FFFF	256KB	<i>Translation Control Unit</i> (TCU) 1 for PCIe root port.
0x00_4F44_0000	0x00_4F45_FFFF	128KB	<i>Translation Buffer Unit</i> (TBU) 0 for PCIe root port.
0x00_4F46_0000	0x00_4F47_FFFF	128KB	<i>Translation Buffer Unit</i> (TBU) 1 for PCIe root port.

5.2.3 Manageability Control Processor memory map

The following figure shows the Morello SDP Manageability Control Processor (MCP) memory map.

Figure 5-3: MCP memory map



The following table shows the Morello SDP MCP memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-3: MCP memory map

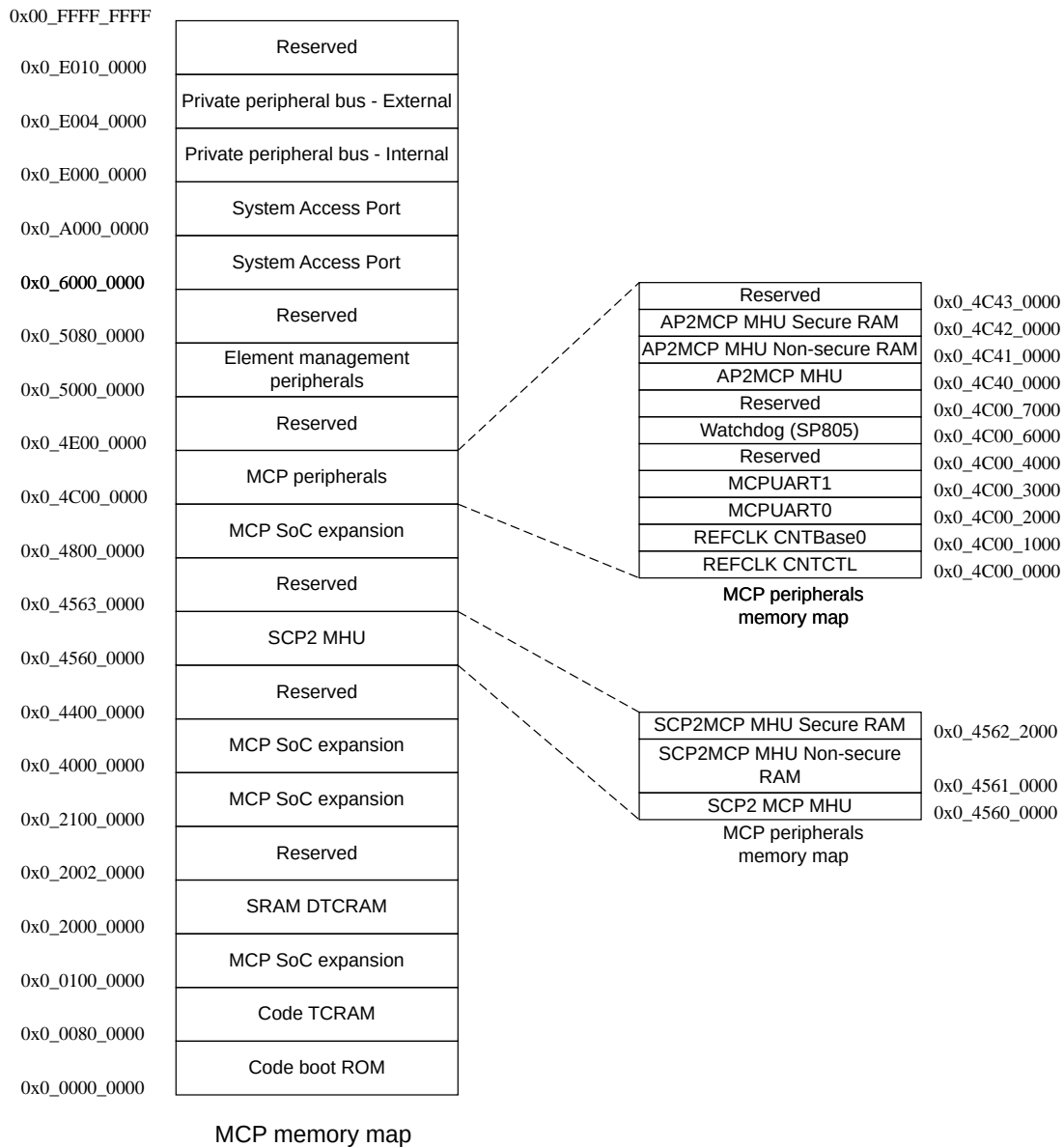
Address range From	Address range To	Size	Description
0x0_0000_0000	0x0_007F_FFFF	8MB	Code boot ROM
0x0_0080_0000	0x0_00FF_FFFF	8MB	Code TCRAM
0x0_0100_0000	0x0_15FF_FFFF	336MB	Reserved part of MCP SoC expansion memory
0x0_1600_0000	0x0_17FF_FFFF	32MB	TMIF interface
0x0_1800_0000	0x0_1FFF_FFFF	128MB	Reserved part of MCP SoC expansion memory
0x0_2000_0000	0x0_2001_FFFF	128KB	SRAM DTCRAM
0x0_2100_0000	0x0_2FFF_FFFF	240MB	Reserved part of MCP SoC expansion memory
0x0_3000_0000	0x0_33FF_FFFF	64MB	MCP QSPI AHB
0x0_3400_0000	0x0_3400_0FFF	4KB	MCP QSPI APB
0x0_3400_1000	0x0_3FFF_DFFF	191MB	Reserved part of MCP SoC expansion memory
0x0_3FFF_E000	0x0_3FFF_EFFF	4KB	Reserved
0x0_3FFF_F000	0x0_3FFF_FFFF	4KB	MCP I2C 1 (BMC-PCC)
0x0_4000_0000	0x0_43FF_FFFF	64MB	MCP SoC expansion
0x0_4560_0000	0x0_45FF_FFFF	10MB	SCP2 MHU
0x0_4800_0000	0x0_4BFF_FFFF	64MB	MCP SoC expansion
0x0_4C00_0000	0x0_4DFF_FFFF	32MB	MCP peripherals
0x0_5000_0000	0x0_507F_FFFF	8MB	Element management peripherals
0x0_6000_0000	0x0_9FFF_FFFF	1GB	System Access Port. Translated to 0x0_4000_0000 to 0x0_7FFF_FFFF of AP memory map.
0x0_A000_0000	0x0_DFFF_FFFF	1GB	System Access Port. Translated to 0x0_0000_0000 to 0x0_3FFF_FFFF of AP memory map with debug address translation not enabled. Translated to 0x4_0000_0000 to 0x4_3FFF_FFFF of AP memory map with debug address translation enabled.
0x0_E000_0000	0x0_E003_FFFF	256KB	Private peripheral bus - Internal.
0x0_E004_0000	0x0_E00F_FFFF	768KB	Private peripheral bus - External.
0x0_E010_0000	0x0_FFFF_FFFF	511MB	Reserved

5.2.4 Manageability Control Processor peripherals memory map

The Manageability Control Processor (MCP) memory map of the Morello SDP contains a region associated with the MCP peripherals.

The following figure shows the peripherals region of the MCP memory map.

Figure 5-4: MCP peripherals memory map



The following table shows the peripherals region of the MCP memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-4: MCP peripherals memory map

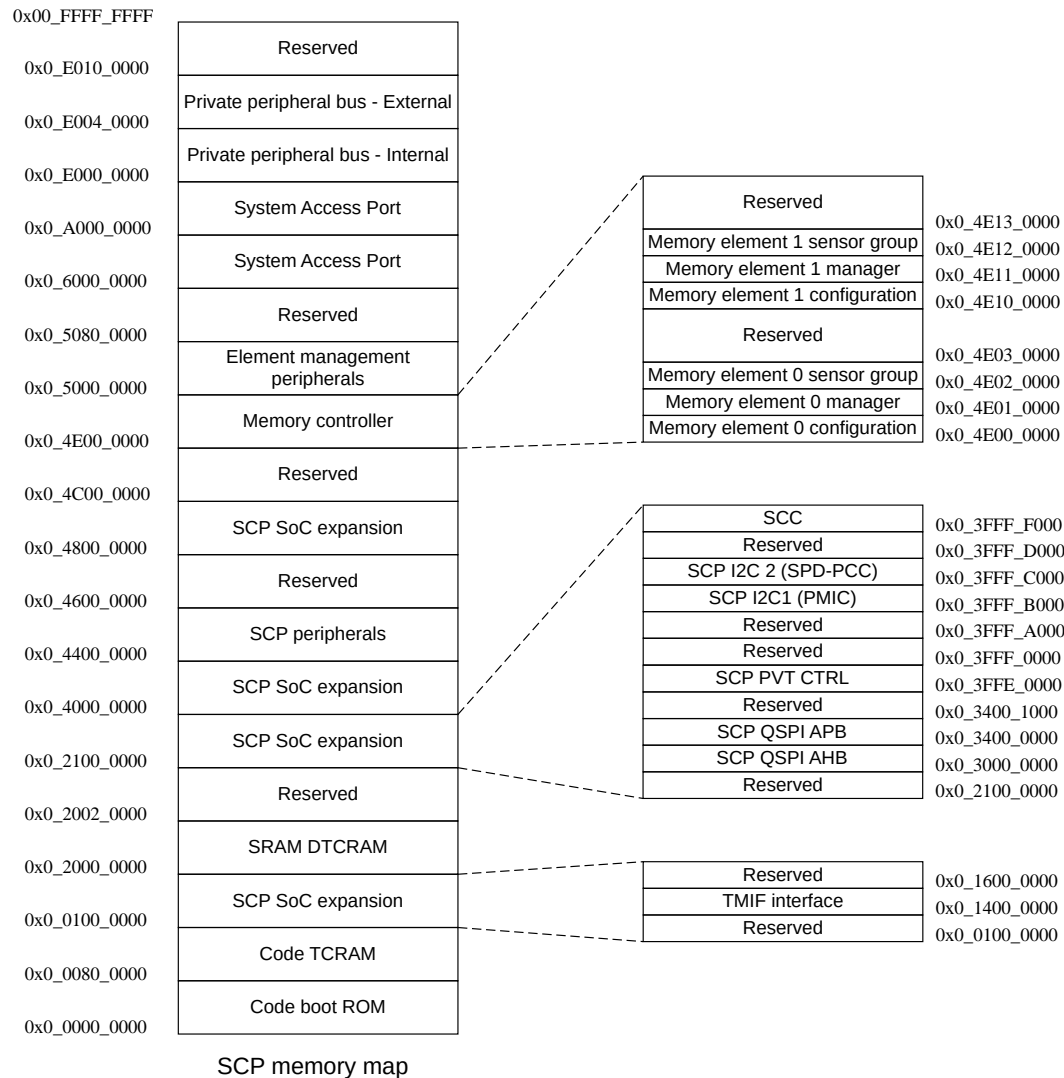
Address range From	Address range To	Size	Description
0x00_4560_0000	0x00_4560_FFFF	64KB	SCP2 MCP <i>Message Handling Unit</i> (MHU)
0x00_4561_0000	0x00_4561_FFFF	64KB	SCP2 MCP MHU Non-secure RAM
0x00_4562_0000	0x00_4562_FFFF	64KB	SCP2 MCP MHU Secure RAM

Address range From	Address range To	Size	Description
0x00_4C00_0000	0x00_4C00_0FFF	4KB	REFCLK CNTCTL
0x00_4C00_1000	0x00_4C00_1FFF	4KB	REFCLK CNTBase0
0x00_4C00_2000	0x00_4C00_2FFF	4KB	MCPUART0
0x00_4C00_3000	0x00_4C00_3FFF	4KB	MCPUART1
0x00_4C00_6000	0x00_4C00_6FFF	4KB	Watchdog (SP805)
0x00_4C40_0000	0x00_4C40_FFFF	64KB	AP2 MCP MHU
0x00_4C41_0000	0x00_4C41_FFFF	64KB	AP2 MCP MHU Non-secure RAM
0x00_4C42_0000	0x00_4C42_FFFF	64KB	AP2 MCP MHU Secure RAM

5.2.5 System Control Processor memory map

The following figure shows the Morello SDP System Control Processor (SCP) memory map.

Figure 5-5: SCP memory map



The following table shows the Morello SDP SCP memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-5: SCP memory map

Address range from	Address range to	Size	Description
0x0_0000_0000	0x0_007F_FFFF	8MB	Code boot ROM
0x0_0080_0000	0x0_00FF_FFFF	8MB	Code TCRAM

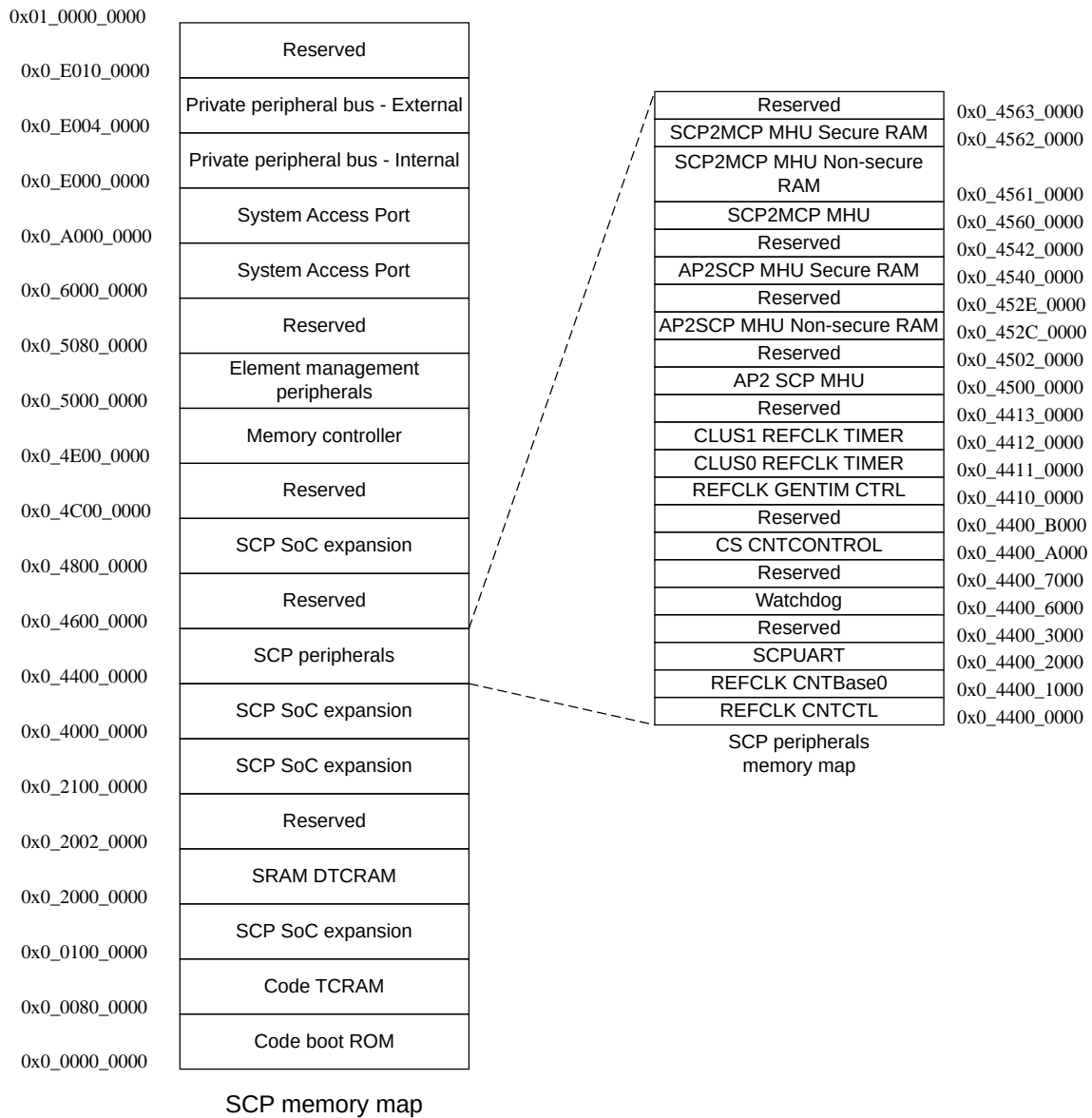
Address range from	Address range to	Size	Description
0x0_0100_0000	0x0_13FF_FFFF	304MB	Reserved part of SCP SoC expansion memory
0x0_1400_0000	0x0_15FF_FFFF	32MB	TMIF interface
0x0_1600_0000	0x0_1FFF_FFFF	160MB	Reserved part of SCP SoC expansion memory
0x0_2000_0000	0x0_2001_FFFF	128KB	SRAM DTCRAM
0x0_2100_0000	0x0_2FFF_FFFF	240MB	Reserved part of SCP SoC expansion memory
0x0_3000_0000	0x0_33FF_FFFF	64MB	SCP QSPI AHB
0x0_3400_0000	0x0_3400_0FFF	4KB	SCP QSPI APB
0x0_3400_1000	0x0_3FFD_FFFF	191MB	Reserved part of SCP SoC expansion memory
0x0_3FFE_0000	0x0_3FFE_FFFF	64KB	SCP PVT CTRL
0x0_3FFF_0000	0x0_3FFF_9FFF	40KB	Reserved part of SCP SoC expansion memory
0x0_3FFF_A000	0x0_3FFF_AFFF	4KB	Reserved
0x0_3FFF_B000	0x0_3FFF_BFFF	4KB	SCP I2C1 (PMIC)
0x0_3FFF_C000	0x0_3FFF_CFFF	4KB	SCP I2C2 (SPD-PCC)
0x0_3FFF_D000	0x0_3FFF_EFFF	8KB	Reserved part of SCP SoC expansion memory
0x0_3FFF_F000	0x0_3FFF_FFFF	4KB	SCC registers
0x0_4000_0000	0x0_43FF_FFFF	64MB	SCP SoC expansion
0x0_4400_0000	0x0_45FF_FFFF	32MB	SCP peripherals
0x0_4800_0000	0x0_4BFF_FFFF	64MB	MCP SoC expansion
0x0_4E00_0000	0x0_4E00_FFFF	64KB	Memory element 0 configuration
0x0_4E01_0000	0x0_4E01_FFFF	64KB	Memory element 0 manager
0x0_4E02_0000	0x0_4E02_FFFF	64KB	Memory element 0 sensor group
0x0_4E10_0000	0x0_4E10_FFFF	64KB	Memory element 1 configuration
0x0_4E11_0000	0x0_4E11_FFFF	64KB	Memory element 1 manager
0x0_4E12_0000	0x0_4E12_FFFF	64KB	Memory element 1 sensor group
0x0_5000_0000	0x0_507F_FFFF	8MB	Element management peripherals
0x0_6000_0000	0x0_9FFF_FFFF	1GB	System Access Port. Translated to 0x0_4000_0000 to 0x0_7FFF_FFFF of AP memory map.
0x0_A000_0000	0x0_DFFF_FFFF	1GB	System Access Port. Translated to 0x0_0000_0000 to 0x0_3FFF_FFFF of AP memory map with debug address translation not enabled. Translated to 0x4_0000_0000 to 0x4_3FFF_FFFF of AP memory map with debug address translation enabled.
0x0_E000_0000	0x0_E003_FFFF	256KB	Private peripheral bus - Internal.
0x0_E004_0000	0x0_E00F_FFFF	768KB	Private peripheral bus - External.
0x0_E010_0000	0x0_FFFF_FFFF	511MB	Reserved

5.2.6 System Control Processor peripherals memory map

The System Control Processor (SCP) memory map of the Morello SDP contains a region associated with the SCP peripherals.

The following figure shows the peripherals region of the SCP memory map.

Figure 5-6: SCP peripherals memory map



The following table shows the peripherals region of the Morello SDP SCP memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-6: SCP peripherals memory map

Address range from	Address range to	Size	Description
0x00_4400_0000	0x00_4400_0FFF	4KB	REFCLK CNTCTL
0x00_4400_1000	0x00_4400_1FFF	4KB	REFCLK CNTBase0
0x00_4400_2000	0x00_4400_2FFF	4KB	SCPUART
0x00_4400_6000	0x00_4400_6FFF	4KB	Watchdog (SP805)

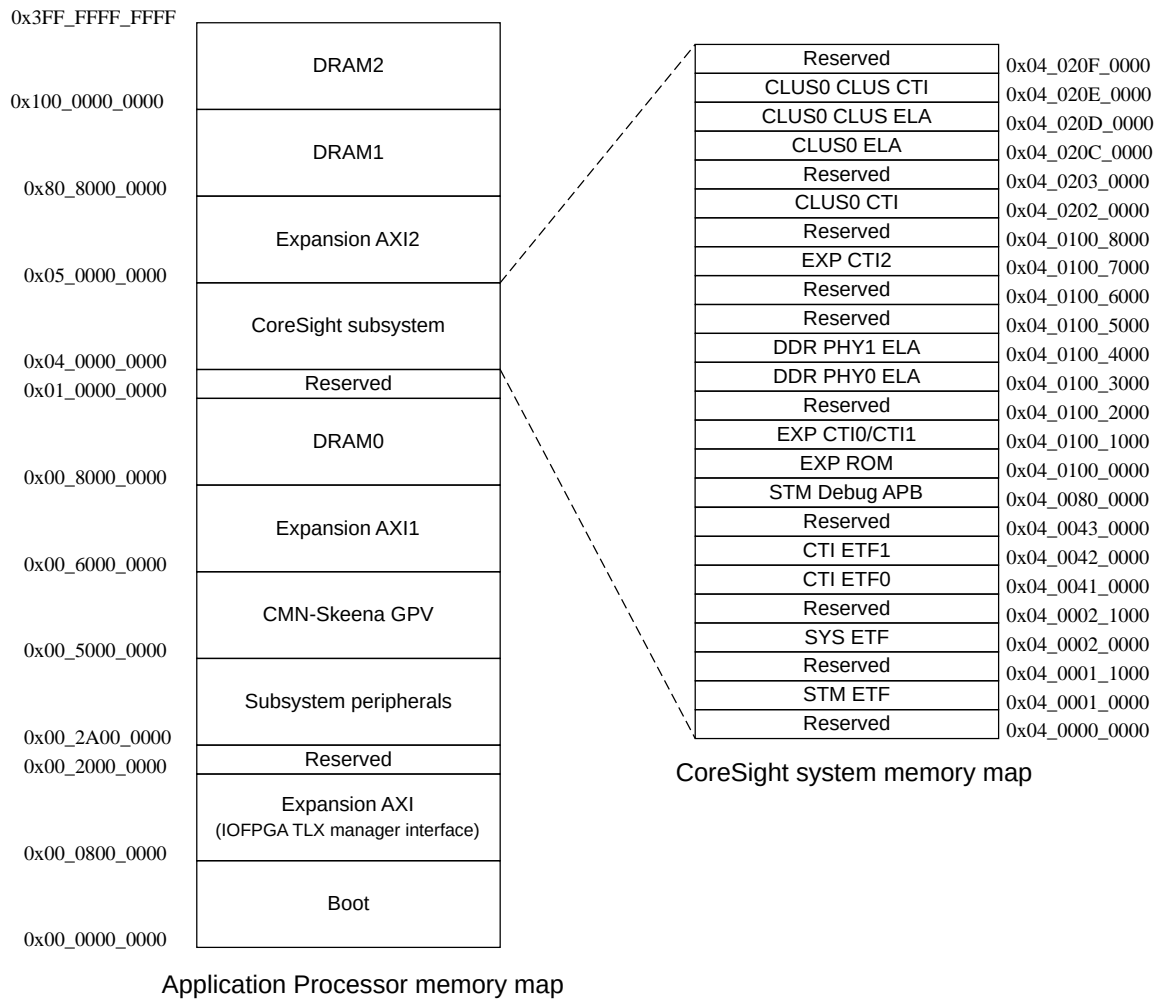
Address range from	Address range to	Size	Description
0x00_4400_A000	0x00_4400_AFFF	4KB	CS CNTCONTROL
0x00_4410_0000	0x00_4410_FFFF	64KB	REFCLK general timer control
0x00_4411_0000	0x00_4411_FFFF	64KB	Cluster 0 time frame
0x00_4412_0000	0x00_4412_FFFF	64KB	Cluster 1 time frame
0x00_4500_0000	0x00_4501_FFFF	128KB	AP2SCP <i>Message Handling Unit</i> (MHU)
0x00_452C_0000	0x00_452D_FFFF	128KB	AP2SCP MHU Non-secure RAM
0x00_4540_0000	0x00_4541_FFFF	128KB	AP2SCP MHU Secure RAM
0x00_4560_0000	0x00_4560_FFFF	64KB	SCP2MCH MHU
0x00_4561_0000	0x00_4561_FFFF	64KB	SCP2MCP MHU Non-secure RAM
0x00_4562_0000	0x00_4562_FFFF	64KB	SCP2MCP MHU Secure RAM

5.2.7 CoreSight system memory map

The Morello SDP Application Processor (AP) memory map contains a region that is associated with the CoreSight™ debug and trace.

The following figure shows the CoreSight™ debug and trace memory map.

Figure 5-7: CoreSight system memory map



The following table shows the peripherals region of the Morello SDP CoreSight™ debug and trace memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-7: CoreSight system memory map

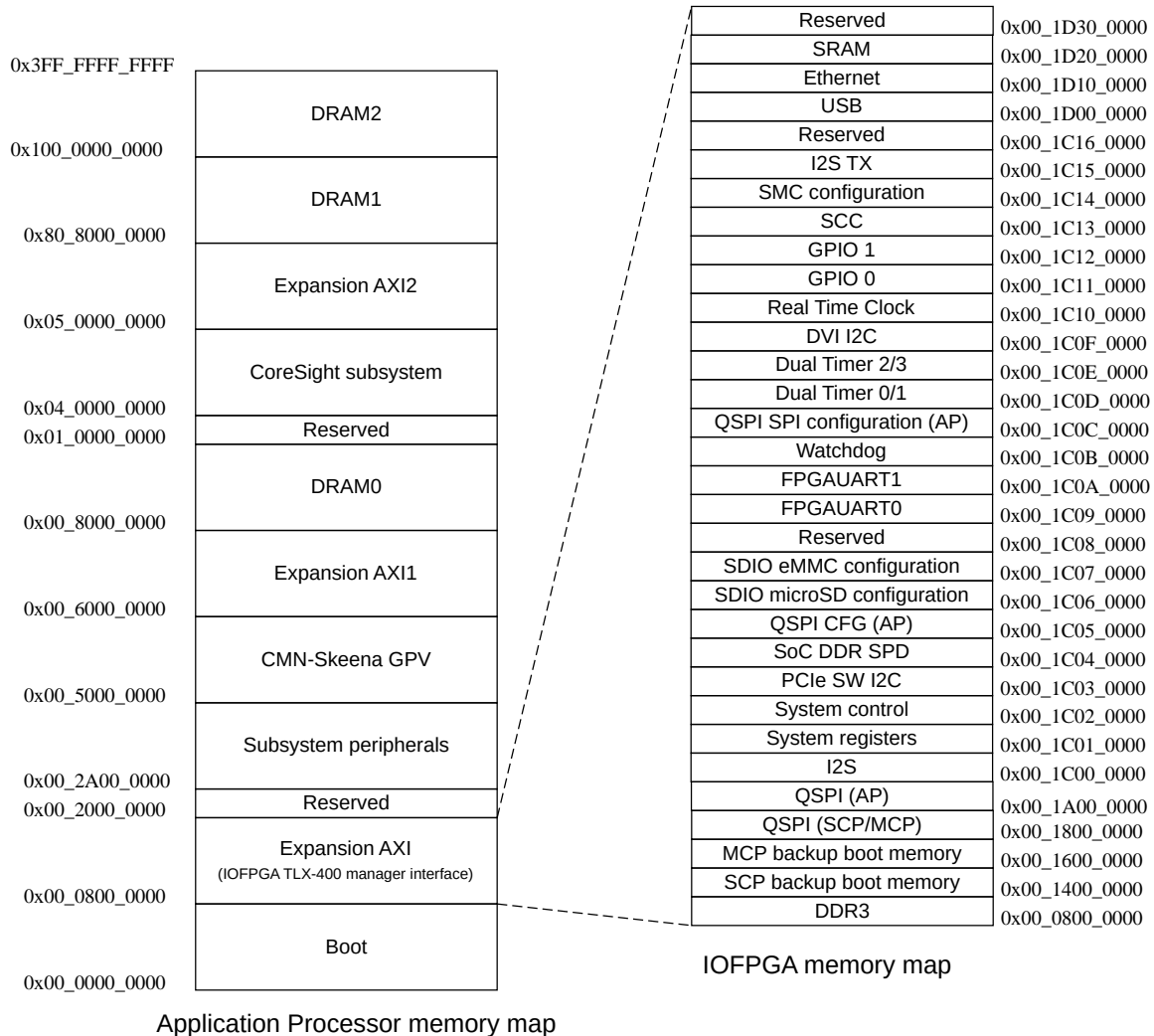
Address range From	Address range To	Size	Description
0x04_0000_0000	0x04_0000_FFFF	64KB	Reserved
0x04_0001_0000	0x04_0001_0FFF	4KB	STM ETF
0x04_0002_0000	0x04_0002_0FFF	4KB	SYS ETF
0x04_0041_0000	0x04_0041_FFFF	64KB	CTI ETF0
0x04_0042_0000	0x04_0042_FFFF	64KB	CTI ETF1
0x04_0080_0000	0x04_00FF_FFFF	4KB	STM Debug APB
0x04_0100_0000	0x04_0100_0FFF	4KB	EXP ROM

Address range From	Address range To	Size	Description
0x04_0100_1000	0x04_0100_1FFF	4KB	EXP CTI0/CTI1
0x04_0100_2000	0x04_0100_2FFF	4KB	Reserved
0x04_0100_3000	0x04_0100_3FFF	4KB	DDR PHY0 ELA
0x04_0100_4000	0x04_0100_4FFF	4KB	DDR PHY1 ELA
0x04_0100_5000	0x04_0100_5FFF	4KB	Reserved
0x04_0100_6000	0x04_0100_6FFF	4KB	Reserved
0x04_0100_7000	0x04_0100_7FFF	4KB	EXP CTI2
0x04_0202_0000	0x04_0202_FFFF	64KB	CLUS0 CTI
0x04_020C_0000	0x04_020C_FFFF	64KB	CLUS0 ELA
0x04_020D_0000	0x04_020D_FFFF	64KB	CLUS0 CLUS ELA
0x04_020E_0000	0x04_020E_FFFF	64KB	CLUS0 CLUS CTI

5.2.8 IOFPGA memory map

The following figure shows the memory map of the peripherals inside the IOFPGA.

Figure 5-8: IOFPGA memory map



The following table shows the IOFPGA memory map. Undefined locations of the memory map are reserved. Software must not attempt to access these locations.

Table 5-8: IOFPGA memory map

Address range From	Address range To	Size	Description
0x0800_0000	0x13FF_FFFF	192MB	DDR3
0x1400_0000	0x15FF_FFFF	32MB	SCP backup boot memory
0x1600_0000	0x17FF_FFFF	32MB	MCP backup boot memory

Address range From	Address range To	Size	Description
0x1800_0000	0x19FF_FFFF	32MB	QSPI (SCP/MCP)
0x1A00_0000	0x1BFF_FFFF	32MB	QSPI (AP)
0x1C00_0000	0x1C00_FFFF	64KB	I ² S
0x1C01_0000	0x1C01_FFFF	64KB	System registers
0x1C02_0000	0x1C02_FFFF	64KB	System control
0x1C03_0000	0x1C03_FFFF	64KB	PCIe SW I2C
0x1C04_0000	0x1C04_FFFF	64KB	SoC DDR SPD
0x1C05_0000	0x1C05_FFFF	64KB	QSPI configuration (AP)
0x1C06_0000	0x1C06_FFFF	64KB	SDIO microSD configuration
0x1C07_0000	0x1C07_FFFF	64KB	SDIO eMMC configuration
0x1C08_0000	0x1C08_FFFF	64KB	Reserved
0x1C09_0000	0x1C09_FFFF	64KB	FPGAUART0
0x1C0A_0000	0x1C0A_FFFF	64KB	FPGAUART1
0x1C0B_0000	0x1C0B_FFFF	64KB	Watchdog
0x1C0C_0000	0x1C0C_FFFF	64KB	QSPI SPI configuration (AP)
0x1C0D_0000	0x1C0D_FFFF	64KB	Dual Timer 0/1
0x1C0E_0000	0x1C0E_FFFF	64KB	Dual Timer 2/3
0x1C0F_0000	0x1C0F_FFFF	64KB	DVI I2C
0x1C10_0000	0x1C10_FFFF	64KB	Real Time Clock
0x1C11_0000	0x1C11_FFFF	64KB	GPIO 0
0x1C12_0000	0x1C12_FFFF	64KB	GPIO 1
0x1C13_0000	0x1C13_FFFF	64KB	SCC
0x1C14_0000	0x1C14_FFFF	64KB	SMC configuration
0x1C15_0000	0x1C15_FFFF	64KB	I ² S TX
0x1D00_0000	0x1D0F_FFFF	1MB	USB
0x1D10_0000	0x1D1F_FFFF	1MB	Ethernet
0x1D20_0000	0x1D2F_FFFF	1MB	SRAM
0x1D30_0000	0x1FFF_FFFF	45MB	Reserved

5.3 Morello SoC interrupt maps

The Morello SoC contains three independent interrupt maps for the *Application Processors* (APs), the *System Control Processor* (SCP), and the *Manageability Control Processor* (MCP).

5.3.1 Application Processor interrupt map

GIC-600 implements two types of interrupt. *Private Peripheral Interrupts* (PPIs) exist separately for each core. *Shared Peripheral Interrupts* (SPIs) are shared between all cores.

The following table shows the *Private Peripheral Interrupts* (PPIs) for the application processors. The map repeats for each core in the subsystem. The *System Control Processor* (SCP) and the *Manageability Control Processor* (MCP) receive interrupts from several sources.

Table 5-9: Private peripheral interrupts

ID	Source	Description
20-16	-	Reserved
21	PMBIRQn	SPE interrupt request
22	COMMIRQn	Debug Communications Channel receive or transmit request
23	PMUIRQn	PMU interrupt
24	CTIIRQ	CTI Interrupt
25	VCPUMNTIRQn	Virtual Maintenance Interrupt (PPI6)
26	CNTHPIRQn	Non-secure PL2 Timer event (PPI5)
27	CNTVIRQn	Virtual Timer event (PPI4)
28	CNTHVIRQn	-
29	CNTPSIRQn	Secure PL1 Physical Timer event (PPI1)
30	CNTPNSIRQn	Non-secure PL1 Physical Timer event (PPI2)
31	-	Reserved

The following table shows the *Shared Peripheral Interrupts* (SPIs) for the application processors.

Table 5-10: Shared peripheral interrupts

ID	Source	Description
32	DMC0_pmuirq	PMU event interrupt from DMC0
33	DMC0_comb_err_overflow	Combined Error interrupt overflow from DMC0
34	DMC0_failed_access_int	TZ access error interrupt from DMC0
35	DMC0_ecc_err	ECC Error from DMC0
36	DMC1_pmuirq	PMU event interrupt from DMC1
37	DMC1_comb_err_overflow	Combined Error interrupt overflow from DMC1
38	DMC1_failed_access_int	TZ access error interrupt from DMC1
39	DMC1_ecc_err	ECC Error from DMC1
66-40	-	Reserved
67	MCP2APMHU_NS	MHU Non-secure interrupt
68	-	Reserved
69	MCP2APMHU_S	MHU secure interrupt
70	-	Reserved
71	ETR	ETRBUFFINT interrupt
72	TCU0_PRI_Q_IRPT_NS	PRI Interrupt from PCIe TCU

ID	Source	Description
73	TCU1_PRI_Q_IRPT_NS	PRI interrupt from CCIX TCU
77-74	-	Reserved
78	CMN600_INTREQPMU_DTC0	PMU Count Overflow Interrupt
82-79	-	Reserved
83	STM-500	STM-500 Synchronization Interrupt
84	CTI	CTI Trigger output 6 from CTI2
85	CTI	CTI Trigger output 7 from CTI2
86	Trusted Watchdog	Trusted Watchdog interrupt(WSO)
87	AP_SEC_UART_INT	AP secure UART interrupt.
90-88	-	Reserved
91	AP_REFCLK Generic Timer (Secure)	AP_REFCLK Generic Timer Interrupt (Secure)
92	AP_REFCLK Generic Timer (Non-secure)	AP_REFCLK Generic Timer Interrupt (Non-secure)
93	Generic Watchdog	Watchdog WSO Interrupt
94	Generic Watchdog	Watchdog WS1 Interrupt
95	AP0 UART	AP UART0 interrupt
96	AP2 UART	AP UART2 interrupt
97	GPU	GPU interrupt request
98	GPU	Job interrupt request
99	GPU	MMU interrupt request
100	GPU	GPU event request (Reserved for Odin/Griff)
101	Display	Interrupt output from Global Control Unit of the DPU
102	Display	Interrupt output from AFBC DMA Unit of the DPU
103	-	Reserved
104	DISPLAY0 TBU	TBU PMU IRPT
105	-	Reserved
106	-	Reserved
107	DISPLAY TCU	Event Queue Secure interrupt, indicating Event Queue Non-Empty or Overflow event_q_irpt_s
108	DISPLAY TCU	Event Queue Non-Secure interrupt, indicating Event Queue Non-Empty or Overflow event_q_irpt_ns
109	DISPLAY TCU	Reserved
110	DISPLAY TCU	SYNC Complete Non-secure Interrupt cmd_sync_irpt_ns
111	DISPLAY TCU	SYNC Complete Secure interrupt cmd_sync_irpt_s
112	DISPLAY TCU	Global Non-secure interrupt global_irpt_ns
113	DISPLAY TCU	Global Secure interrupt global_irpt_s
114	-	Reserved
115	DISPLAY TCU	pmu_irpt PMU (Performance Monitor Unit) interrupt
125-116	-	Reserved
126	CLUS0_PMUIRQn	DSU PMU interrupt
127	CLUS0_PMUIRQn	DSU PMU interrupt

ID	Source	Description
128	Morello board	AP external IRQ (AP_EXT_INT)
129	Morello board	AP external Ethernet IRQ (AP_EXT_ETHERNET_INT)
167-130	-	Reserved
168	Morello SoC	GPIO IOFPGA combined IRQ
176-169	Morello SoC	GPIO IOFPGA combined IRQ [7:0]
200-177	-	Reserved
201	Morello SoC	pcie_inta_out
202	Morello SoC	pcie_intb_out
203	Morello SoC	pcie_intc_out
204	Morello SoC	pcie_intd_out
205	Morello SoC	pcie_phy_interrupt_out
206	Morello SoC	pcie_aer_interrupt
207	Morello SoC	pcie_link_down_reset_out
208	Morello SoC	pcie_local_interrupt_reset
209	Morello SoC	pcie_performance_data_threshold
210	Morello SoC	pcie_negotiated_speed_change
211	Morello SoC	pcie_link_training_done
212	Morello SoC	pcie_pll_status_rise
213	Morello SoC	pcie_message_fifo_interrupt
214	Morello SoC	pcie_local_interrupt_ras
215	Morello SoC	pcie_phy_lane_interrupt
231-216	-	Reserved
232	Morello SoC	ccix_bus_device_change_irq
233	Morello SoC	ccix_inta_out
234	Morello SoC	ccix_intb_out
235	Morello SoC	ccix_intc_out
236	Morello SoC	ccix_intd_out
237	Morello SoC	ccix_phy_interrupt_out
238	Morello SoC	ccix_aer_interrupt
239	Morello SoC	ccix_link_down_reset_out
240	Morello SoC	ccix_local_interrupt_reset
241	Morello SoC	ccix_performance_data_threshold
242	Morello SoC	ccix_negotiated_speed_change
243	Morello SoC	ccix_link_training_done
244	Morello SoC	ccix_pll_status_rise
245	Morello SoC	ccix_message_fifo_interrupt
246	Morello SoC	ccix_local_interrupt_ras
247	Morello SoC	ccix_hot_reset_irq
248	Morello SoC	ccix_flr_reset_irq
249	Morello SoC	ccix_power_state_change_irq

ID	Source	Description
250	Morello SoC	ccix_phy_lane_interrupt
255-251	-	Reserved
256	MMUTCU1_PMU_IRPT	PMU interrupt
257	MMUTCU1_EVENT_Q_IRPT_S	Event Queue Secure interrupt, indicating Event Queue Non-Empty or Overflow
258	MMUTCU1_CMD_SYNC_IRPT_S	SYNC Complete Secure interrupt
259	MMUTCU1_GLOBAL_IRPT_S	Global Secure interrupt
260	MMUTCU1_EVENT_Q_IRPT_NS	Event Queue non-Secure interrupt, indicating Event Queue Non-Empty or Overflow
261	MMUTCU1_CMD_SYNC_IRPT_NS	SYNC Complete Non-secure interrupt
262	MMUTCU1_GLOBAL_IRPT_NS	Global Non-secure interrupt
263	MMUTCU2_PMU_IRPT	PMU interrupt
264	MMUTCU2_EVENT_Q_IRPT_S	Event Queue Secure interrupt, indicating Event Queue Non-Empty or Overflow
265	MMUTCU2_CMD_SYNC_IRPT_S	SYNC Complete Secure interrupt
266	MMUTCU2_GLOBAL_IRPT_S	Global Secure interrupt
267	MMUTCU2_EVENT_Q_IRPT_NS	Event Queue non-Secure interrupt, indicating Event Queue Non-Empty or Overflow
268	MMUTCU2_CMD_SYNC_IRPT_NS	SYNC Complete Non-secure interrupt
269	MMUTCU2_GLOBAL_IRPT_NS	Global Non-secure interrupt
319-270	-	Reserved
323-320	MMUTBU_PMU_IRPT[3:0]	TBU PMU Interrupt. Allocated to 4 TBUs in the system.
511-324	-	Reserved
512	CLUSTER0SCP ->AP MHU Non-secure	-
513	CLUSTER0SCP ->AP MHU secure	-
514	CLUSTER1SCP ->AP MHU Non-secure	-
515	CLUSTER1SCP ->AP MHU secure	-
575-516	-	Reserved
576	PO_REFCLK_GENTIM	Pn_REFCLK Generic Secure Timer interrupts
577	PO_REFCLK_GENTIM	Pn_REFCLK Generic Secure Timer interrupts
640-578	-	Reserved

5.3.2 System Control Processor interrupt map

The *System Control Processor* (SCP) receives interrupts from several sources.

The sources of the interrupts to the SCP are:

- Application Processor system wakeup interrupts
- CoreSight™ power and reset request interrupts
- Internal SCP subsystem interrupts
- Expansion SCP interrupts

The interrupts are routed to the Nested Vector Interrupt Controllers in the Cortex®-M7 processors where they can be managed by software.

The following table shows the SCP interrupts.

Table 5-11: SCP interrupts

ID	Source	Description
NMI	SCP Generic Watchdog	SCP Watchdog (WS0)
0	-	Reserved
1	CoreSight	CoreSight debug power up request (If there is a separate debug power domain). Note: SCP Firmware must support optional debug power up rail.
2	CoreSight	CoreSight system power up request
3	CoreSight	CoreSight debug reset request
4	GIC expansion interrupt	External GIC wakeup interrupt. Generated by the logical OR of all the GIC Expansion Interrupts.
15-5	-	Reserved
16	SCP external IRQ (SCP_EXT_INT)	SCP external IRQ (SCP_EXT_INT)
17	GPIO pads	GPIO-IOFPGA combined IRQ
25-18	GPIO pads	GPIO-IOFPGA individual IRQ[7:0] and IOFPGA external IRQ[7:0]
32-26	-	Reserved
33	SCP REFCLK Generic Timer	REFCLK Physical Timer interrupt
34	GENTIM_SYNC	System generic timer synchronization interrupt
35	CSTS_SYNC	CoreSight Time stamp synchronization interrupt
36	-	Reserved
37	CTI	CTI Trigger 0
38	CTI	CTI Trigger 1
39	GICECCFATAL	GIC Fatal ECC failure
40	GICAXIMERR	GIC Fatal AXI manager error
41	-	Reserved
42	AON_UART_INT	Always-on UART interrupt
43	-	Reserved
44	Generic Watchdog	Generic Watchdog timer interrupt WS0
45	Generic Watchdog	Generic Watchdog timer interrupt WS1
46	Trusted Watchdog	Trusted Watchdog timer interrupt WS0
47	Trusted Watchdog	Trusted Watchdog timer interrupt WS1
48	APPS_UART_INT	Applications UART interrupt
49	-	Reserved
50	CPU Core Power Policy Units	Consolidated CPU PPU Interrupt for cores
53-51	-	Reserved
54	CPU Cluster Power Policy Units	Consolidated CPU cluster PPU Interrupt for clusters 0-1
55	CPU Core PLLs	Consolidated CPU PLL Lock for PLLs

ID	Source	Description
58-56	-	Reserved
59	CPU Core Fault Indicator	Consolidated nFaultIRQ for both clusters
63-60	-	Reserved
64	CPU ECC error interrupts	Consolidated nERRIRQ for both clusters
68-64	-	Reserved
69	Cluster PLLs	Consolidated lock interrupt for cluster PLLs
70	Cluster PLLs	Consolidated unlock interrupt for cluster PLLs
71	dso_clu0_irq0	Cluster0 DSO interrupt0
72	dso_clu0_irq1	Cluster0 DSO interrupt1
73	dso_clu1_irq0	Cluster1 DSO interrupt0
74	dso_clu1_irq1	Cluster1 DSO interrupt1
81-75	-	Reserved
82	AP2SCP MHU Non-secure interrupt	Consolidated MHU High Priority Non-Secure interrupt
83	AP2SCP MHU Secure interrupt	Consolidated MHU Secure interrupt for both clusters
84	MCP2SCP MHU Non-secure Interrupt	MCP2SCP MHU High Priority Interrupt
85	MCP2SCP MHU Secure Interrupt	MCP2SCP MHU High Priority Interrupt
89-86	-	Reserved
90	Pn_REFCLK_GENTIM_1_2	Consolidated Pn REFCLK Timer Interrupt for both clusters
93-91	-	Reserved
94	CONS_MMU_TCU_RASIRPT	Consolidated MMU RAS for the interrupt coming from multiple TCUs
95	CONS_MMU_TBU_RAS IRPT[NUM_TBUS-1:0]	Consolidated TBU for the interrupts coming from various TBUs
96	INTREQPPU	PPU interrupt from CMN-Ske
97	INTREQERRNS	Non Secure error handling interrupt from CMN-Skeena
98	INTREQERRS	Secure error handling interrupt from CMN-Skeena
99	INTREQFAULTS	Secure Fault handling interrupt from CMN-Skeena
100	INTREQFAULTNS	Non Secure Fault handling interrupt from CMN-Skeena
101	INTREQPMU	PMU count overflow interrupt
119-102	-	Reserved
127-120	DBGCH [0-7]_PPU_INT	-
129	-	Reserved
130	Power Integration Kit	Debug PIK Interrupt
131	LOGIC_PPU_INT	LOGIC_PPU_INT
134-132	-	Reserved
135	SRAM_PPU_INT	SRAM PPU Interrupt
136	DPU_PPU_INT	Display PPU Interrupt
137	GPU_PPU_INT	GPU PPU Interrupt
138	-	Reserved
139	MCP WS1	MCP Watchdog reset
140	SYSPLL_LOCK	Sys PLL Lock
141	SYSPLL_UNLOCK	Sys PLL Unlock

ID	Source	Description
142	INTPLL_LOCK	Interconnect PLL Lock
143	INTPLL_UNLOCK	Interconnect PLL UnLock
144	DPUPLL_LOCK	DISPLAY PLL Lock
145	DPUPLL_UNLOCK	DISPLAY PLL UnLock
146	GPUPLL_LOCK	GPU PLL Lock
147	GPUPLL_UNLOCK	GPU PLL UnLock
148	PXLPLL_LOCK	Pixel PLL Lock
149	PXLPLL_UNLOCK	Pixel PLL UnLock
173-150	-	Reserved
174	DMC_PLL_LOCK	DMC PLL Lock
175	DMC_PLL_UNLOCK	DMC PLL UnLock
179-176	-	Reserved
180	DMC0 Interrupt	DMC0_misc oflow
181	DMC0 Interrupt	DMC0_err_oflow
182	DMC0 Interrupt	DMC0_ecc_err_int
183	DMC0 Interrupt	DMC0_misc_access_int
184	DMC0 Interrupt	DMC0_temperature_event_int
185	DMC0 Interrupt	DMC0_failed_access_int
186	DMC0 Interrupt	DMC0_mgr_int
187	DMC1 Interrupt	DMC1_misc oflow
188	DMC1 Interrupt	DMC1_err_oflow
189	DMC1 Interrupt	DMC1_ecc_err_int
190	DMC1 Interrupt	DMC1_misc_access_int
191	DMC1 Interrupt	DMC1_temperature_event_int
192	DMC1 Interrupt	DMC1_failed_access_int
193	DMC1 Interrupt	DMC1_mgr_int
208	-	Reserved
209	SCP PMIC I2C	SCP PMIC I2C interrupt (I2C1)
210	SCP SPD-PCC I2C	SCP SPD-PCC I2C interrupt (I2C2)
211	SCP-QSPI	SCP QSPI interrupt
212	PVT controller	PVT controller interrupt
218-213	-	Reserved
219	ccix_bus_device_change_irq	ccix_bus_device_change_irq
220	ccix_inta_out	ccix_inta_out
221	ccix_intb_out	ccix_intb_out
222	ccix_intc_out	ccix_intc_out
223	ccix_intd_out	ccix_intd_out
224	ccix_phy_interrupt_out	ccix_phy_interrupt_out
225	ccix_aer_interrupt	ccix_aer_interrupt
226	ccix_lnk_down_reset_out	ccix_lnk_down_reset_out

ID	Source	Description
227	ccix_lo cal_interrupt_reset	ccix_lo cal_interrupt_reset
228	ccix_perform ance_data_threshold	ccix_perform ance_data_threshold
229	ccix_nego tiated_speed_change	ccix_nego tiated_speed_change
230	ccix_link_training_done	ccix_link_training_done
231	cc ix_pll_status_rise	cc ix_pll_status_rise
232	ccix_mes sage_fifo_interrupt	ccix_mes sage_fifo_interrupt
233	ccix_local_interrupt_ras	ccix_local_interrupt_ras
234	ccix_hot_reset_irq	ccix_hot_reset_irq
235	ccix_flr_reset_irq	ccix_flr_reset_irq
236	ccix_powe r_state_change_irq	ccix_powe r_state_change_irq
237	pcie_aer_interrupt	pcie_aer_interrupt
238	pcie_lo cal_interrupt_reset	pcie_lo cal_interrupt_reset
239	pcie_local_interrupt_ras	pcie_local_interrupt_ras

5.3.3 Manageability Control Processor interrupt map

The *Manageability Control Processor* (MCP) receives interrupts from several sources.

The sources of the interrupts to the MCP are:

- Application Processor system wakeup interrupts
- CoreSight™ power and reset request interrupts
- Internal MCP subsystem interrupts
- Expansion MCP interrupts

The interrupts are routed to the Nested Vector Interrupt Controllers in the Cortex®-M7 processors where they can be managed by software.

The following table shows the MCP interrupts.

Table 5-12: MCP interrupts

ID	Source	Description
NMI	MCP Generic Watchdog	MCP Watchdog (WS0)
0	-	Reserved
1	CoreSight	CoreSight debug power up request (If there is a separate debug power domain). Note: MCP Firmware must support optional debug power up rail.
2	CoreSight	CoreSight system power up request
3	CoreSight	CoreSight debug reset request
4	GIC expansion interrupt	External GIC wakeup interrupt. Generated by the logical OR of all the GIC Expansion Interrupts.
15-5	-	Reserved

ID	Source	Description
16	MCP external IRQ (MCP_EXT_INT)	MCP external IRQ (MCP_EXT_INT)
17	GPIO pads	GPIO-IOFPGA combined IRQ
25-18	GPIO pads	GPIO-IOFPGA individual IRQ[7:0] and IOFPGA external IRQ[7:0]
32-26	-	Reserved
33	MCP REFCLK Generic Timer	REFCLK Physical Timer interrupt
34	Non-secure AP2MCP MHU	MHU Non-Secure interrupt
35	-	Reserved
36	AP2MCP Secure MHU	MHU Secure interrupt
37	CTI	CTI Trigger 0
38	CTI	CTI Trigger 1
41-39	-	Reserved
42	MCP_UART0_INT	Always-on UART interrupt
83-43	MCP_UART1_INT	Always-on UART interrupt
84	MCP2SCP MHU Non-secure Interrupt	MCP2SCP MHU High Priority Interrupt
85	MCP2SCP MHU Secure Interrupt	MCP2SCP MHU High Priority Interrupt
93-86	-	Reserved
94	MMU_TBU_RASIRPT[NUM_TBUS-1:0]	Consolidated MMU RAS for the interrupt coming from multiple TCUs
95	MMU_TBU_RASIRPT[NUM_TBUS-1:0]	Consolidated TBU for the interrupts coming from multiple TBUs
96	INTREQPPU	PPU interrupt from CMN-Skeena
97	INTREQERRNS	Non-secure error handling interrupt from CMN-Skeena
98	INTREQERRS	Secure error handling interrupt from CMN-Skeena
99	INTREQFAULTS	Secure Fault handling interrupt from CMN-Skeena
100	INTREQFAULTNS	Non-secure Fault handling interrupt from CMN-Skeena
101	INTREQPMU	PMU count overflow interrupt
138-102	-	Reserved
139	MCP WS1	MCP Watchdog reset
179-140	SYSPLL_LOCK	Sys PLL Lock
180	DMC0 Interrupt	DMC0_misc oflow
181	DMC0 Interrupt	DMC0_err_oflow
182	DMC0 Interrupt	DMC0_ecc_err_int
183	DMC0 Interrupt	DMC0_misc_access_int
184	DMC0 Interrupt	DMC0_temperature_event_int
185	DMC0 Interrupt	DMC0_failed_access_int
186	DMC0 Interrupt	DMC0_mgr_int
187	DMC1 Interrupt	DMC1_misc oflow
188	DMC1 Interrupt	DMC1_err_oflow
189	DMC1 Interrupt	DMC1_ecc_err_int
190	DMC1 Interrupt	DMC1_misc_access_int
191	DMC1 Interrupt	DMC1_temperature_event_int
192	DMC1 Interrupt	DMC1_failed_access_int

ID	Source	Description
193	DMC1 Interrupt	DMC1_mgr_int
208	-	Reserved
209	MCP BMC-PCC I2C	MCP BMC-PCC I2C interrupt (I2C1)
210	MCP-QSPI	MCP QSPI interrupt
219-211	-	Reserved
220	pcie_aer_interrupt	pcie_aer_interrupt
221	pcie_local_interrupt_reset	pcie_local_interrupt_reset
222	pcie_local_interrupt_ras	pcie_local_interrupt_ras
223	ccix_aer_interrupt	ccix_aer_interrupt
224	ccix_local_interrupt_reset	ccix_phy_interrupt_out
225	ccix_local_interrupt_ras	ccix_aer_interrupt
239-226	ccix_link_down_reset_out	ccix_link_down_reset_out

5.4 System Security Control registers

The *System Security Control* (SSC) interface in the Morello SoC controls system-wide security features.

These features include the following:

- Selection of an internal sources for Debug Authentication signals
- General Purpose register for secure state storage

5.4.1 System Security Control registers summary

The base memory address of the SSC registers is 0x0_2A42_0000 in the subsystem peripherals region of the *Application Processor* (AP) memory map.

The following table shows the SSC registers in offset order from the base memory address. Undefined registers are reserved. Software must not attempt to access these registers.

Table 5-13: SSC registers summary

Offset	Name	Type	Reset	Width	Description
0x0010	SSC_DBG_CFG_STAT	RO	0x0001_0000	32	See SSC_DBG_CFG_STAT Register .
0x0014	SSC_DBG_CFG_SET	WO	N/A	32	See SSC_DBG_CFG_SET Register .
0x0018	SSC_DBG_CFG_CLR	WO	N/A	32	See SSC_DBG_CFG_CLR Register .
0x0028	SSC_AUXDBG_CFG	RW	0x0000_0000	32	See SSC_AUXDBG_CFG Register .
0x0030	SSC_GPRETN	RW	0x0000_0000	32	See SSC_GPRETN Register .
0x0040	SSC_VERSION	RO	0x1004_17B3	32	See SSC_VERSION Register .
0x0500	SSC_CHIPID_ST	RO	0x0000_0000	32	See SSC_CHIPID_ST Register .
0x0FD0	SSC_PID4	RO	0x0000_0004	32	See SSC_PID4 Register .

Offset	Name	Type	Reset	Width	Description
0x0FE0	SSC_PID0	RO	0x0000_0044	32	See SSC_PID0 Register .
0x0FE4	SSC_PID1	RO	0x0000_00B8	32	See SSC_PID1 Register .
0x0FE8	SSC_PID2	RO	0x0000_000B	32	See SSC_PID2 Register .
0x0FF0	COMPID0	RO	0x0000_000D	32	See SSC_COMPID0 Register .
0x0FF4	COMPID1	RO	0x0000_00F0	32	See SSC_COMPID1 Register .
0x0FF8	COMPID2	RO	0x0000_0005	32	See SSC_COMPID2 Register .
0x0FFC	COMPID3	RO	0x0000_00B1	32	See SSC_COMPID3 Register .

5.4.2 SSC_DBGCFG_STAT Register

The SSC_DBGCFG_STAT Register characteristics are:

Purpose

Controls how the Debug Authentication signals are to be driven, either from an external source, or internally using build-in register bits, also implemented using this register.

Defines the values of the Debug Authentication signals when they are configured to be internally driven.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_DBGCFG_STAT Register bit assignments.

Table 5-14: SSC_DBGCFG_STAT Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7]	SPIDEN_SEL_STAT	RO	<p>Selects between SPIDEN external or internal drive:</p> <p>0b0: External.</p> <p>0b1: Internal.</p> <p>If external mode is selected SPIDEN is driven by top-level configuration input SPIDEN_CFG.</p> <p>Reset value 0b0.</p>
[6]	SPIDEN_INT_STAT	RO	<p>SPIDEN internal drive value.</p> <p>Reset value 0b0.</p>

Bits	Name	Type	Function
[5]	SPNIDEN_SEL_STAT	RO	<p>Selects between SPNIDEN external or internal drive:</p> <p>0b0: External.</p> <p>0b1: Internal.</p> <p>If external mode is selected SPNIDEN is driven by top-level configuration input.</p> <p>Reset value 0b0.</p>
[4]	SPNIDEN_INT_STAT	RO	<p>SPNIDEN internal drive value.</p> <p>Reset value 0b0.</p>
[3]	DEVICEEN_SEL_STAT	RO	<p>Selects between DEVICEEN external or internal drive:</p> <p>0b0: External.</p> <p>0b1: Internal.</p> <p>If external mode is selected DEVICEEN is driven by top-level configuration input.</p> <p>Reset value 0b0.</p>
[2]	DEVICEEN_INT_STAT	RO	<p>DEVICEEN internal drive value.</p> <p>Reset value 0b0.</p>
[1:0]	-	-	Reserved.

5.4.3 SSC_DBGCFG_SET Register

The SSC_DBGCFG_SET Register characteristics are:

Purpose

The SSC_DBGCFG_SET register is a Secure access only write-only memory mapped register. This register is associated with the SSC_DBGCFG_STAT register. Writing 0b1 to a particular field in the SSC_DBGCFG_SET register sets the corresponding bit in the SSC_DBGCFG_STAT register to 1.

Usage constraints

This register is write-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_DBGCFG_SET Register bit assignments.

Table 5-15: SSC_DBGCFG_SET Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7]	SPIDEN_SEL_SET	RO	Sets SPIDEN_SEL_STAT to 0b1: 0b0: No effect. 0b1: Set SPIDEN_SEL_STAT to 0b1.
[6]	SPIDEN_INT_SET	RO	Sets SPIDEN_INT_STAT to 0b1: 0b0: No effect. 0b1: Set SPIDEN_INT_STAT to 0b1.
[5]	SPNIDEN_SEL_SET	RO	Sets SPNIDEN_SEL_STAT to 0b1: 0b0: No effect. 0b1: Set SPNIDEN_INT_STAT to 0b1.
[4]	SPNIDEN_INT_SET	RO	Sets SPNIDEN_INT_STAT to 0b1: 0b0: No effect. 0b1: Set SPNIDEN_INT_STAT to 0b1.
[3]	DEVI CEEN_SEL_SET	RO	Sets DEVI CEEN_SEL_STAT to 0b1: 0b0: No effect. 0b1: Set DEVI CEEN_SEL_STAT to 0b1.
[2]	DEVI CEEN_INT_SET	RO	Sets DEVI CEEN_INT_STAT to 0b1: 0b0: No effect. 0b1: Set DEVI CEEN_INT_STAT to 0b1.
[1:0]	-	-	Reserved.

5.4.4 SSC_DBGCFG_CLR Register

The SSC_DBGCFG_CLR Register characteristics are:

Purpose

The SSC_DBGCFG_CLR register is a Secure access only write-only memory mapped register. This register is associated with the SSC_DBGCFG_STAT register. Writing 0b1 to a particular field in the SSC_DBGCFG_CLR register clears the corresponding bit in the SSC_DBGCFG_STAT register to 0.

Usage constraints

This register is write-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_DBGCFG_CLR Register bit assignments.

Table 5-16: SSC_DBGCFG_CLR Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7]	SPIDEN_SEL_CLR	WO	Clears SPIDEN_SEL_STAT to 0b1: 0b0: No effect. 0b1: Clear SPIDEN_SEL_STAT to 0b0.
[6]	SPIDEN_INT_CLR	WO	Clears SPIDEN_INT_STAT to 0b1: 0b0: No effect. 0b1: Clear SPIDEN_INT_STAT to 0b0.
[5]	SPNIDEN_SEL_CLR	WO	Clears SPNIDEN_SEL_STAT to 0b1: 0b0: No effect. 0b1: Clear SPNIDEN_INT_STAT to 0b0.
[4]	SPNIDEN_INT_CLR	WO	Clears SPNIDEN_INT_STAT to 0b1: 0b0: No effect. 0b1: Clear SPNIDEN_INT_STAT to 0b0.
[3]	DEVICEEN_SEL_CLR	WO	Clears DEVICEEN_SEL_STAT to 0b1: 0b0: No effect. 0b1: Clear DEVICEEN_SEL_STAT to 0b0.
[2]	DEVICEEN_INT_CLR	WO	Clears DEVICEEN_INT_STAT to 0b0: 0b0: No effect. 0b1: Clear DEVICEEN_INT_STAT to 0b0.
[1:0]	-	-	Reserved.

5.4.5 SSC_AUXDBGCFG Register

The SSC_AUXDBGCFG Register characteristics are:

Purpose

The SSC_AUXDBGCFG register is a Secure access only read-write register. The register provides override control of the debug authentication signals **DBGEN** and **NIDEN**.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_AUXDBGCFG Register bit assignments.

Table 5-17: SSC_AUXDBGCFG Register bit assignments

Bits	Name	Type	Function
[31:2]	-	-	Reserved.
[1:0]	INTERNAL_DEBUG_OVERRIDE	RW	<p>0b00: Enable Non-secure self-hosted debug. DBGEN and NIDEN inputs to the application processors are HIGH.</p> <p>0b01: Disable Invasive, Non-secure self-hosted debug.</p> <p>Enable Non-invasive, Non-secure self-hosted debug.</p> <p>DBGEN inputs to the application processors are LOW and NIDEN inputs to the application processors are HIGH.</p> <p>0b1: Disable Non-secure self-hosted debug.</p> <p>DBGEN and NIDEN inputs to the application processors are LOW.</p> <p>Reset value 0b00.</p>



Arm strongly recommends that this register is not used, and that you leave both bits at their reset value.

5.4.6 SSC_GPRETN Register

The SSC_GPRETN Register characteristics are:

Purpose

The SSC_GPRETN register is a Secure access read/write memory mapped register that provides 16 bit general storage for security purposes. The register resets only on system powerup reset.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_GPRETN Register bit assignments.

Table 5-18: SSC_GPRETN Register bit assignments

Bits	Name	Type	Function
[31:16]	-	-	Reserved.
[15:0]	GPRETN	RW	General-purpose register for Secure state storage. Reset value 0x0000.

5.4.7 SSC_VERSION Register

The SSC_VERSION Register characteristics are:

Purpose

The SSC_VERSION register is a Secure access read-only memory mapped register that specifies the Morello SoC version ID for security purposes.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_VERSION Register bit assignments.

Table 5-19: SSC_VERSION Register bit assignments

Bits	Name	Type	Function
[31:28]	CONFIGURATION	RO	Equals 0b0001 on Morello SoC.

Bits	Name	Type	Function
[27:24]	MAJOR_REVISION	RO	Equals 0b0000 on Morello SoC.
[23:20]	MINOR_REVISION	RO	Equals 0b0000 on Morello SoC.
[19:12]	DESIGNER_ID	RO	Equals Arm identifier 0x41 on Morello SoC.
[11:0]	PART_NUMBER	RO	Equals Arm identifier 0x7B3 on Morello SoC.

5.4.8 SSC_SW_SCRATCH Registers

The SSC_SW_SCRATCH Register characteristics are:

Purpose

The SSC_SW_SCRATCH registers are scratch registers for use by software.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_SW_SCRATCH Register bit assignments.

Table 5-20: SSC_SW_SCRATCH Register bit assignments

Bits	Name	Type	Function
[31:0]	SCRATCH	RW	Software scratch values. Reset value 0x0000_0000.

5.4.9 SSC_SW_CAP Registers

The SSC_SW_CAP Register characteristics are:

Purpose

The SSC_SW_CAP registers are capability registers used by the System Control Processor (SCP) software to record the design configuration.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_SW_CAP Register bit assignments.

Table 5-21: SSC_SW_CAP Register bit assignments

Bits	Name	Type	Function
[31:0]	SW_CAP	RW	SCP software defined capability registers. Reset value 0x0000_0000

5.4.10 SSC_SW_CAPCTRL Register

The SSC_SW_CAPCTRL Register characteristics are:

Purpose

The SSC_SW_CAPCTRL register contains a stick bit to enable writing to the SSC_SW_CAP registers.

Usage constraints

Once set, the active bit, bit[0] cannot be cleared until SoC_nPOR is asserted.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_SW_CAPCTRL Register bit assignments.

Table 5-22: SSC_SW_CAPCTRL Register bit assignments

Bits	Name	Type	Function
[31:1]	-	-	Reserved.
[0]	SW_CAP_WR_EN	RO	Sticky bit to enable writing to SSC_SW_CAP registers. 0: Enable writes. 1: Disable writes. Once set to 1, this bit cannot be cleared until SoC_nPOR is asserted. Reset value 0.

5.4.11 SSC_CHIPID_ST Register

The SSC_CHIPID_ST Register characteristics are:

Purpose

The SSC_CHIPID_ST register stores the CHIPID status for the node when there are multiple sockets.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_CHIPID_ST Register bit assignments.

Table 5-23: SSC_CHIPID_ST Register bit assignments

Bits	Name	Type	Function
[31:9]	-	-	Reserved.
[8]	MU_LTI_CHIP_MODE	RO	Multi-chip mode tie-off value. 0: Single chip. 1: Multi-chip. Reset value 0b0.
[7:6]	-	-	Reserved.
[5:0]	CHIP_ID	RO	Tie-off value in multi-chip mode. This is 0b0 for single chip mode. Reset value 0b000000

5.4.12 SSC_PID4 Register

The SSC_PID4 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_PID4 Register bit assignments.

Table 5-24: SSC_PID4 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:4]	SIZE	RO	LOG2 of the number of 4KB blocks occupied by the interface. Reset value 0x0.
[3:0]	DES_2	RO	JEP106 continuation code to identify designer. Reset value 0x4 for Arm.

5.4.13 SSC_PID0 Register

The SSC_PID0 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_PID0 Register bit assignments.

Table 5-25: SSC_PID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	PART_0	RO	Bits [7:0] of part number. Reset value 0x44.

5.4.14 SSC_PID1 Register

The SSC_PID1 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_PID1 Register bit assignments.

Table 5-26: SSC_PID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:4]	DES_0	RO	Bits[3:0] of JEP identity. Reset value 0xB.

Bits	Name	Type	Function
[3:0]	PART_1	RO	Bits[11:8] of part number. Reset value 0x8.

5.4.15 SSC_PID2 Register

The SSC_PID2 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_PID2 Register bit assignments.

Table 5-27: SSC_PID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:4]	REVISION	RO	Revision number. Reset value 0x0.
[3]	JEDEC	RO	JEDEC ID. Reset value 0b1.
[2:0]	DES_1	RO	Designer ID. Reset value 0b011.

5.4.16 SSC_COMPID0 Register

The SSC_COMPID0 Register characteristics are:

Purpose

The SSC_COMPID0 register stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_COMPID0 Register bit assignments.

Table 5-28: SSC_COMPID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	SSC_COMPID0	RO	Component ID 0 information. Reset value 0x0D.

5.4.17 SSC_COMPID1 Register

The SSC_COMPID1 Register characteristics are:

Purpose

The SSC_COMPID1 register stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_COMPID1 Register bit assignments.

Table 5-29: SSC_COMPID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	SSC_COMPID1	RO	Component ID 1 information. Reset value 0xF0.

5.4.18 SSC_COMPID2 Register

The SSC_COMPID2 Register characteristics are:

Purpose

The SSC_COMPID2 register stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_COMPID2 Register bit assignments.

Table 5-30: SSC_COMPID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	SSC_COMPID2	RO	Component ID 2 information. Reset value 0x05.

5.4.19 SSC_COMPID3 Register

The SSC_COMPID3 Register characteristics are:

Purpose

The SSC_COMPID3 register stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [System Security Control registers summary](#).

The following table shows the SSC_COMPID3 Register bit assignments.

Table 5-31: SSC_COMPID3 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	SSC_COMPID3	RO	Component ID 3 information. Reset value 0xB1.

5.5 Serial Configuration Control registers

The System Configuration Control (SCC) registers contain the initial settings of blocks before bootup. Write and read accesses to these registers during run-time enable software to alter and to read block settings.

5.5.1 Serial Configuration Control registers summary

The base memory address of the SCC registers in the Morello SDP is `0x0_3FFF_F000` in the *System Control Processor* (SCP) SoC expansion region of the SCP memory map.

The following table shows the SCC registers in offset order from the base memory address. Undefined registers are reserved. Software must not attempt to access these registers.

Table 5-32: SCC registers summary

Offset	Name	Type	Reset	Width	Description
0x0004	PMCLK_DIV	RW/RO	0x0001_0001	32	See PMCLK_DIV Register.
0x000C	SYSAPBCLK_CTRL	RW/RO	0x0000_0202	32	See SYSAPBCLK_CTRL Register.
0x0010	SYSAPBCLK_DIV	RW/RO	0x0013_0013	32	See SYSAPBCLK_DIV Register.
0x0018	IOFPGA_TMIF2XCLK_CTRL	RW/RO	0x0000_0202	32	See IOFPGA_TMIF2XCLK_CTRL Register.
0x001C	IOFPGA_TMIF2XCLK_DIV	RW/RO	0x0000_0000	32	See IOFPGA_TMIF2XCLK_DIV Register.
0x0024	IOFPGA_TSIF2XCLK_CTRL	RW/RO	0x0000_0202	32	See IOFPGA_TSIF2XCLK_CTRL Register.
0x0028	IOFPGA_TSIF2XCLK_DIV	RW/RO	0x000B_000B	32	See IOFPGA_TSIF2XCLK_DIV Register.
0x0030	SCPNICCLK_CTRL	RW/RO	0x0000_0202	32	See SCPNICCLK_CTRL Register.
0x0034	SCPNICCLK_DIV	RW/RO	0x0000_0000	32	See SCPNICCLK_DIV Register.
0x003C	SCPI2CCLK_CTRL	RW/RO	0x0000_0202	32	See SCPI2CCLK_CTRL Register.
0x0040	SCPI2CCLK_DIV	RW/RO	0x000F_000F	32	See SCPI2CCLK_DIV Register.
0x0048	SCPQSPICLK_CTRL	RW/RO	0x0000_0202	32	See SCPQSPICLK_CTRL Register.
0x004C	SCPQSPICLK_DIV	RW/RO	0x0000_0000	32	See SCPQSPICLK_DIV Register.
0x0054	SENSORCLK_CTRL	RW/RO	0x0000_0202	32	See SENSORCLK_CTRL Register.
0x0058	SENSORCLK_DIV	RW/RO	0x0017_0017	32	See SENSORCLK_DIV Register.
0x0060	MCPNICCLK_CTRL	RW/RO	0x0000_0202	32	See MCPNICCLK_CTRL Register.
0x0064	MCPNICCLK_DIV	RW/RO	0x0000_0000	32	See MCPNICCLK_DIV Register.
0x006C	MCPI2CCLK_CTRL	RW/RO	0x0000_0202	32	See MCPI2CCLK_CTRL Register.
0x0070	MCPI2CCLK_DIV	RW/RO	0x0017_0017	32	See MCPI2CCLK_DIV Register.
0x0078	MCPQSPICLK_CTRL	RW/RO	0x0000_0202	32	See MCPQSPICLK_CTRL Register.
0x007C	MCPQSPICLK_DIV	RW/RO	0x0000_0000	32	See MCPQSPICLK_DIV Register.
0x0084	PCIEAXICLK_CTRL	RW/RO	0x0000_0202	32	See PCIEAXICLK_CTRL Register.
0x0088	PCIEAXICLK_DIV	RW/RO	0x0001_0001	32	See PCIEAXICLK_DIV Register.
0x0090	CCIXAXICLK_CTRL	RW/RO	0x0000_0101	32	See CCIXAXICLK_CTRL Register.
0x0094	CCIXAXICLK_DIV	RW/RO	0x0001_0001	32	See CCIXAXICLK_DIV Register.
0x009C	PCIEAPBCLK_CTRL	RW/RO	0x0000_0101	32	See PCIEAPBCLK_CTRL Register.
0x00A0	PCIEAPBCLK_DIV	RW/RO	0x000B_000B	32	See PCIEAPBCLK_DIV Register.
0x00A8	CCIXAPBCLK_CTRL	RW/RO	0x0000_0101	32	See CCIXAPBCLK_CTRL Register.
0x00AC	CCIXAPBCLK_DIV	RW/RO	0x000B_000B	32	See CCIXAPBCLK_DIV Register.
0x00F0	SYS_CLK_EN	RW	0x0000_3FF7	32	See SYS_CLK_EN Register.
0x0100	CPU0_PLL_CTRL0	RW	0x8010_3000	32	See CPU0_PLL_CTRL0 Register.
0x0104	CPU0_PLL_CTRL1	RW/RO	0x9100_0000	32	See CPU0_PLL_CTRL1 Register.

Offset	Name	Type	Reset	Width	Description
0x0108	CPU1_PLL_CTRL0	RW	0x8010_3000	32	See CPU1_PLL_CTRL0 Register .
0x010C	CPU1_PLL_CTRL1	RW/RO	0x9100_0000	32	See CPU1_PLL_CTRL1 Register .
0x0110	CLUS_PLL_CTRL0	RW	0x8010_2000	32	See CLUS_PLL_CTRL0 Register .
0x0114	CLUS_PLL_CTRL1	RW/RO	0x9100_0000	32	See CLUS_PLL_CTRL1 Register .
0x0118	SYS_PLL_CTRL0	RW	0x8010_3000	32	See SYS_PLL_CTRL0 Register .
0x011C	SYS_PLL_CTRL1	RW/RO	0x9100_0000	32	See SYS_PLL_CTRL1 Register .
0x0120	DMC_PLL_CTRL0	RW	0x8020_2000	32	See DMC_PLL_CTRL0 Register .
0x0124	DMC_PLL_CTRL1	RW/RO	0x9100_0000	32	See DMC_PLL_CTRL1 Register .
0x0128	INT_PLL_CTRL0	RW	0x8010_2000	32	See INT_PLL_CTRL0 Register .
0x012C	INT_PLL_CTRL1	RW/RO	0x9100_0000	32	See INT_PLL_CTRL1 Register .
0x0130	GPU_PLL_CTRL0	RW	-	32	See GPU_PLL_CTRL0 Register .
0x0134	GPU_PLL_CTRL1	RW/RO	-	32	See GPU_PLL_CTRL1 Register .
0x0138	DPU_PLL_CTRL0	RW	-	32	See DPU_PLL_CTRL0 Register .
0x013C	DPU_PLL_CTRL1	RW/RO	-	32	See DPU_PLL_CTRL1 Register .
0x0140	PXL_PLL_CTRL0	RW	-	32	See PXL_PLL_CTRL0 Register .
0x0144	PXL_PLL_CTRL1	RW/RO	-	32	See PXL_PLL_CTRL1 Register .
0x0150	SYS_MAN_RESET	RW	0x0000_0C00	32	See SYS_MAN_RESET Register .
0x0160	BOOT_CTL	RW/RO	0x0000_0000	32	See BOOT_CTL Register .
0x0164	BOOT_CTRL_STA	RW/RO	0x0000_0000	32	See BOOT_CTRL_STA Register .
0x0168	SCP_BOOT_ADR	RW/RO	0x0000_0000	32	See SCP_BOOT_ADR Register .
0x016C	MCP_BOOT_ADR	RW/RO	0x0000_0000	32	See MCP_BOOT_ADR Register .
0x0170	PLATFORM_CTRL	RW/RO	0x0000_0000	32	See PLATFORM_CTRL Register .
0x0174	TARGETIDAPP	RW/RO	0x07B3_0477	32	See TARGETIDAPP Register .
0x0178	TARGETIDSCP	RW/RO	0x07B5_0477	32	See TARGETIDSCP Register .
0x017C	TARGETIDMCP	RW/RO	0x07B4_0477	32	See TARGETIDMCP Register .
0x0180	BOOT_GPR0	RW/RO	0x0000_0000	32	See BOOT_GPR0 Register .
0x0184	BOOT_GPR1	RW/RO	0x0000_0000	32	See BOOT_GPR1 Register .
0x0188	BOOT_GPR2	RW/RO	0x0000_0000	32	See BOOT_GPR2 Register .
0x018C	BOOT_GPR3	RW/RO	0x0000_0000	32	See BOOT_GPR3 Register .
0x0190	BOOT_GPR4	RW/RO	0x0000_0000	32	See BOOT_GPR4 Register .
0x0194	BOOT_GPR5	RW/RO	0x0000_0000	32	See BOOT_GPR5 Register .
0x0198	BOOT_GPR6	RW/RO	0x0000_0000	32	See BOOT_GPR6 Register .
0x019C	BOOT_GPR7	RW/RO	0x0000_0000	32	See BOOT_GPR7 Register .
0x01A0	INSTANCE_ID	RW/RO	0x0000_0000	32	See INSTANCE_ID Register .
0x01A4	PCIE_BOOT_CTRL	RW	0x0000_0003	32	See PCIE_BOOT_CTRL Register .
0x01AC	GPU_CTRL	RW	0x0000_0018	32	See GPU_CTRL Register .
0x01B4	DBG_AUTHN_CTRL	RW	0x0000_0007	32	See DBG_AUTHN_CTRL Register .
0x01B8	CTI0_CTRL	RW	0x0000_0000	32	See CTI0_CTRL Register .
0x01BC	CTI1_CTRL	RW	0x0000_0000	32	See CTI1_CTRL Register .
0x01C0	CTI0TO3_CTRL	RW	0x0000_0000	32	See CTI0TO3_CTRL Register .

Offset	Name	Type	Reset	Width	Description
0x01C4	MCP_WDOGCTI_CTRL	RW	0x0000_0000	32	See MCP_WDOGCTI_CTRL Register .
0x01C8	SCP_WDOGCTI_CTRL	RW	0x0000_0000	32	See SCP_WDOGCTI_CTRL Register .
0x01CC	DBGEXPCTI_CTRL	RW	0x0000_0000	32	See DBGEXPCTI_CTRL Register .
0x01D0	PCIE_PM_CTRL	RW/RO	0x0000_0000	32	See PCIE_PM_CTRL Register .
0x01D4	CCIX_PM_CTRL	RW/RO	0x0000_0000	32	See CCIX_PM_CTRL Register .
0x01D8	SCDBG_CTRL	RW/RO	0x0000_0000	32	See SCDBG_CTRL Register .
0x01DC	EXP_IF_CTRL	RW	0x0000_0000	32	See EXP_IF_CTRL Register .
0x01E4	RO_CTRL	RW	0x0000_0001	32	See RO_CTRL Register .
0x01E8	CMN_CCIX_CTRL	RW/RO	0x0101_0000	32	See CMN_CCIX_CTRL Register .
0x01EC	STM_CTRL	RW	0x0000_0000	32	See STM_CTRL Register .
0x01F0	AXI_OVRD_PCIE	RW	0x0030_3030	32	See AXI_OVRD_PCIE Register .
0x01F4	AXI_OVRD_CCIX	RW	0x0030_3030	32	See AXI_OVRD_CCIX Register .
0x01F8	AXI_OVRD_TSIF	RW	0x0000_3030	32	See AXI_OVRD_TSIF Register .
0x01FC	GPU_TEXFMTENABLE	RW	0x0000_0000	32	See GPU_TEXFMTENABLE Register .
0x0200	TRACE_PAD_CTRL0	RW	0x1111_1111	32	See TRACE_PAD_CTRL0 Register .
0x0204	TRACE_PAD_CTRL1	RW	0x0000_1111	32	See TRACE_PAD_CTRL1 Register .
0x0208	IOFPGA_TMIF_PAD_CTRL	RW	0x0011_1111	32	See IOFPGA_TMIF_PAD_CTRL Register .
0x020C	IOFPGA_TSIF_PAD_CTRL	RW	0x0011_1111	32	See IOFPGA_TSIF_PAD_CTRL Register .
0x0210	DISPLAY_PAD_CTRL0	RW	0x0101_0101	32	See DISPLAY_PAD_CTRL0 Register .
0x0214	DISPLAY_PAD_CTRL1	RW	0x0101_0101	32	See DISPLAY_PAD_CTRL1 Register .
0x0E00	APB_CTRL_CLR	RO	0x0000_0000	32	See APB_CTRL_CLR Register .
0x0FD0	PID4	RO	0x0000_0004	32	See PID4 Register .
0xFE0	PID0	RO	0x0000_00AF	32	See PID0 Register .
0xFE4	PID1	RO	0x0000_00B0	32	See PID1 Register .
0xFE8	PID2	RO	0x0000_000B	32	See PID2 Register .
0xFEC	PID3	RO	0x0000_0000	32	See PID3 Register .
0xFF0	CID0	RO	0x0000_000D	32	See CID0 Register .
0xFF4	CID1	RO	0x0000_00F0	32	See CID1 Register .
0xFF8	CID2	RO	0x0000_0005	32	See CID2 Register .
0xFFC	CID3	RO	0x0000_00B1	32	See CID3 Register .

5.5.2 PMCLK_DIV Register

The PMCLK_DIV Register characteristics are:

Purpose

Controls the **PMCLK** division value from **REFCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PMCLK_DIV Register bit assignments.

Table 5-33: PMCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKSEL_CUR+1. Reset value 0b00001, division value=2.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKSEL_CUR+1. Reset value 0b00001, division value=2.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.3 SYSAPBCLK_CTRL Register

The SYSAPBCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **SYSAPBCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SYSAPBCLK_CTRL Register bit assignments.

Table 5-34: SYSAPBCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for SYSAPBCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for SYSAPBCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.4 SYSAPBCLK_DIV Register

The SYSAPBCLK_DIV Register characteristics are:

Purpose

Controls the **SYSAPBCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SYSAPBCLK_DIV Register bit assignments.

Table 5-35: SYSAPBCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value =CLKDIV_CUR+1.
[15:5]	-	-	Reserved.

Bits	Name	Type	Function
[4:0]	CLKDIV	RW	<p>Sets clock division value.</p> <p>Division value = CLKDIV_CUR + 1.</p> <p>Reset value 0b10011, division value = 20.</p>



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.5 IOFPGA_TMIF2XCLK_CTRL Register

The IOFPGA_TMIF2XCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **TMIF2XCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the IOFPGA_TMIF2XCLK_CTRL Register bit assignments.

Table 5-36: IOFPGA_TMIF2XCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	<p>Current value of source for TMIF2XCLK:</p> <p>0b0001: Source is REFCLK.</p> <p>0b0010: Source is divided SYSPLLCLK.</p>
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	<p>Select source for TMIF2XCLK:</p> <p>0b0001: Select REFCLK.</p> <p>0b0010: Select divided SYSPLLCLK.</p> <p>Reset value 0b0010.</p>



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.6 IOFPGA_TMIF2XCLK_DIV Register

The IOFPGA_TMIF2XCLK_DIV Register characteristics are:

Purpose

Controls the TMIF2XCLK division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the IOFPGA_TMIF2XCLK_DIV Register bit assignments.

Table 5-37: IOFPGA_TMIF2XCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b10011, division value=20.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.7 IOFPGA_TSIF2XCLK_CTRL Register

The IOFPGA_TSIF2XCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **TSIF2XCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the IOFPGA_TSIF2XCLK_CTRL Register bit assignments.

Table 5-38: IOFPGA_TSIF2XCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for TSIF2XCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for TSIF2XCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.8 IOFPGA_TSIF2XCLK_DIV Register

The IOFPGA_TSIF2XCLK_DIV Register characteristics are:

Purpose

Controls the TSIF2XCLK division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the IOFPGA_TSIF2XCLK_DIV Register bit assignments.

Table 5-39: IOFPGA_TSIF2XCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b10011, division value=20.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.9 SCPNICCLK_CTRL Register

The SCPNICCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **SCPNICCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCPNICCLK_CTRL Register bit assignments.

Table 5-40: SCPNICCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for SCPNICCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for SCPNICCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.10 SCPNICCLK_DIV Register

The SCPNICCLK_DIV Register characteristics are:

Purpose

Controls the **SCPNICCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCPNICCLK_DIV Register bit assignments.

Table 5-41: SCPNICCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.

Bits	Name	Type	Function
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b00111, division value=8.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.11 SCPI2CCLK_CTRL Register

The SCPI2CCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **SCPI2CCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCPI2CCLK_CTRL Register bit assignments.

Table 5-42: SCPI2CCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for SCPI2CCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RO	Select source for SCPI2CCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0001.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.12 SCPI2CCLK_DIV Register

The SCPI2CCLK_DIV Register characteristics are:

Purpose

Controls the **SCPI2CCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCPI2CCLK_DIV Register bit assignments.

Table 5-43: SCPI2CCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1 Reset value 0b00000, division value = 1.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.13 SCPQSPICLK_CTRL Register

The SCPQSPICLK_CTRL Register characteristics are:

Purpose

Selects source for clock **SCPQSPICLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCPQSPICLK_CTRL Register bit assignments.

Table 5-44: SCPQSPICLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for SCPQSPICLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for SCPQSPICLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0001.



Note

The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.14 SCPQSPICLK_DIV Register

The SCPQSPICLK_DIV Register characteristics are:

Purpose

Controls the **SCPQSPICLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCPQSPICLK_DIV Register bit assignments.

Table 5-45: SCPQSPICLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b00000, division value=1.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.15 SENSORCLK_CTRL Register

The SENSORCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **SENSORCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SENSORCLK_CTRL Register bit assignments.

Table 5-46: SENSORCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for SENSORCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for SENSORCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.16 SENSORCLK_DIV Register

The SENSORCLK_DIV Register characteristics are:

Purpose

Controls the **SENSORCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SENSORCLK_DIV Register bit assignments.

Table 5-47: SENSORCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.

Bits	Name	Type	Function
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b10111, division value = 24.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.17 MCPNICCLK_CTRL Register

The MCPNICCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **MCPNICCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCPNICCLK_CTRL Register bit assignments.

Table 5-48: MCPNICCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for MCPNICCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for MCPNICCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.18 MCPNICCLK_DIV Register

The MCPNICCLK_DIV Register characteristics are:

Purpose

Controls the **MCPNICCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCPNICCLK_DIV Register bit assignments.

Table 5-49: MCPNICCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b00111, division value = 8.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.19 MCPI2CCLK_CTRL Register

The MCPI2CCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **MCPI2CCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCPI2CCLK_CTRL Register bit assignments.

Table 5-50: MCPI2CCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for MCPI2CCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for MCPI2CCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0001.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.20 MCPI2CCLK_DIV Register

The MCPI2CCLK_DIV Register characteristics are:

Purpose

Controls the **MCPI2CCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCPI2CCLK_DIV Register bit assignments.

Table 5-51: MCPI2CCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b00000, division value = 1.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.21 MCPQSPICLK_CTRL Register

The MCPQSPICLK_CTRL Register characteristics are:

Purpose

Selects source for clock **MCPQSPICLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCPQSPICLK_CTRL Register bit assignments.

Table 5-52: MCPQSPICLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for MCPQSPICLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for MCPQSPICLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0001.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.22 MCPQSPICLK_DIV Register

The MCPQSPICLK_DIV Register characteristics are:

Purpose

Controls the MCPQSPICLK division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCPQSPICLK_DIV Register bit assignments.

Table 5-53: MCPQSPICLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.

Bits	Name	Type	Function
[4:0]	CLKDIV	RW	<p>Sets clock division value.</p> <p>Division value = CLKDIV_CUR+1.</p> <p>Reset value 0b00000, division value = 1.</p>



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.23 PCIEAXICLK_CTRL Register

The PCIEAXICLK_CTRL Register characteristics are:

Purpose

Selects source for clock **PCIEAXICLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PCIEAXICLK_CTRL Register bit assignments.

Table 5-54: PCIEAXICLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	<p>Current value of source for PCIEAXICLK:</p> <p>0b0001: Source is REFCLK.</p> <p>0b0010: Source is divided SYSPLLCLK.</p>
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	<p>Select source for PCIEAXICLK:</p> <p>0b0001: Select REFCLK.</p> <p>0b0010: Select divided SYSPLLCLK.</p> <p>Reset value 0b0010.</p>



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.24 PCIEAXICK_DIV Register

The PCIEAXICK_DIV Register characteristics are:

Purpose

Controls the **PCIEAXICK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PCIEAXICK_DIV Register bit assignments.

Table 5-55: PCIEAXICK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b00001, division value = 2.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.25 CCIXAXICLK_CTRL Register

The CCIXAXICLK_CTRL Register characteristics are:

Purpose

Selects source for clock **CCIXAXICLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CCIXAXICLK_CTRL Register bit assignments.

Table 5-56: CCIXAXICLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for CCIXAXICLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for CCIXAXICLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.26 CCIXAXICLK_DIV Register

The CCIXAXICLK_DIV Register characteristics are:

Purpose

Controls the **CCIXAXICLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CCIXAXICLK_DIV Register bit assignments.

Table 5-57: CCIXAXICLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b00001, division value = 2.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.27 PCIEAPBCLK_CTRL Register

The PCIEAPBCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **PCIEAPBCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PCIEAPBCLK_CTRL Register bit assignments.

Table 5-58: PCIEAPBCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for PCIEAPBCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for PCIEAPBCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.28 PCIEAPBCLK_DIV Register

The PCIEAPBCLK_DIV Register characteristics are:

Purpose

Controls the **PCIEAPBCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PCIEAPBCLK_DIV Register bit assignments.

Table 5-59: PCIEAPBCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.

Bits	Name	Type	Function
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b01011, division value = 12.



Note

The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.29 CCIXAPBCLK_CTRL Register

The CCIXAPBCLK_CTRL Register characteristics are:

Purpose

Selects source for clock **CCIXAPBCLK**.

Usage constraints

Bits[11:8] are read-only. Bits[3:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CCIXAPBCLK_CTRL Register bit assignments.

Table 5-60: CCIXAPBCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:8]	CLKSEL_CUR	RO	Current value of source for CCIXAPBCLK : 0b0001: Source is REFCLK . 0b0010: Source is divided SYSPLLCLK .
[7:4]	-	-	Reserved.
[3:0]	CLKSEL	RW	Select source for CCIXAPBCLK : 0b0001: Select REFCLK . 0b0010: Select divided SYSPLLCLK . Reset value 0b0010.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.30 CCIXAPBCLK_DIV Register

The CCIXAPBCLK_DIV Register characteristics are:

Purpose

Controls the **CCIXAPBCLK** division value from **SYSPLLCLK**.

Usage constraints

Bits[20:16] are read-only. Bits[4:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CCIXAPBCLK_DIV Register bit assignments.

Table 5-61: CCIXAPBCLK_DIV Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Current clock divider value. Division value = CLKDIV_CUR+1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Sets clock division value. Division value = CLKDIV_CUR+1. Reset value 0b01011, division value = 12.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.31 SYS_CLK_EN Register

The SYS_CLK_EN Register characteristics are:

Purpose

Enables or disables internally generated clocks.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SYS_CLK_EN Register bit assignments.

Table 5-62: SYS_CLK_EN Register bit assignments

Bits	Name	Type	Function
[31:14]	-	-	Reserved.
[13]	CCIXAPBCLKEN	RW	Enable clock CCIXAPBCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
12	CCIXAXICLKEN	RW	Enable clock CCIXAXICLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
11	PCIEAPBCLKEN	RW	Enable clock PCIEAPBCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
10	PCIEAXICLKEN	RW	Enable clock PCIEAXICLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.

Bits	Name	Type	Function
9	MCPQSPICLKEN	RW	Enable clock MCPQSPICLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
8	MCPI2CCLKEN	RW	Enable clock MCPI2CCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
7	MCPNICCLKEN	RW	Enable clock MCPNICCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
6	SENSORCLKEN	RW	Enable clock SENSORCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
5	SCPQSPICLKEN	RW	Enable clock SCPQSPICLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
4	SCPI2CCLKEN	RW	Enable clock SCPI2CCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
3	-	-	Reserved.
2	IOFP_GA_TSIF2XCLKEN	RW	Enable clock TSIF2XCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.

Bits	Name	Type	Function
1	IOFP_GA_TMIF2XCLKEN	RW	Enable clock TMIF2XCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.
0	SYSAPBCLKEN	RW	Enable clock SYSAPBCLK : 0b0: Clock disabled. 0b1: Clock enabled. Reset value 0b1.

5.5.32 CPU0_PLL_CTRL0 Register

The CPU0_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register CPU0_PLL_CTRL1, control the settings of clock control PLL CPU0PLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CPU0_PLL_CTRL0 Register bit assignments.

Table 5-63: CPU0_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 0b1: 0x0: PLL disabled. 0x1: PLL disabled. Reset value 0x1.
[30:26]	-	-	Reserved.
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0b1.

Bits	Name	Type	Function
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x30 , division value=48.
[7:1]	-	-	Reserved.
[0]	HARD_BYPASS	RW	Bypasses PLL to drive input clock directly into the SoC: 0x0: PLL not bypassed. 0x1: PLL bypassed. Reset value 0b0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.33 CPU0_PLL_CTRL1 Register

The CPU0_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register CPU0_PLL_CTRL0, control the settings of clock control PLL CPU0PLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CPU0_PLL_CTRL1 Register bit assignments.

Table 5-64: CPU0_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	Second post-divide value. Post-divide value=POSTDIV2. Reset value 0b1.
[27]	-	-	Reserved.

Bits	Name	Type	Function
[26:24]	POSTDIV1	RW	First post-divide value. Post-divide value=POSTDIV1. Reset value 0b1.
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction =FRAC/2 ²⁴ . Reset value 0x0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.34 CPU1_PLL_CTRL0 Register

The CPU1_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register CPU1_PLL_CTRL1, control the settings of clock control PLL CPU1PLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CPU1_PLL_CTRL0 Register bit assignments.

Table 5-65: CPU1_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 0b1: 0x0: PLL disabled. 0x1: PLL enabled . Reset value 0x1.
[30:26]	-	-	Reserved.
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0b1.

Bits	Name	Type	Function
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x30 , division value=48.
[7:1]	-	-	Reserved.
[0]	HARD_BYPASS	RW	Bypasses PLL to drive input clock directly into the SoC: 0x0: PLL not bypassed. 0x1: PLL bypassed. Reset value 0b0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.35 CPU1_PLL_CTRL1 Register

The CPU1_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register CPU1_PLL_CTRL0, control the settings of clock control PLL CPU1PLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CPU1_PLL_CTRL1 Register bit assignments.

Table 5-66: CPU1_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	Second post-divide value. Post-divide value=POSTDIV2. Reset value 0b1.
[27]	-	-	Reserved.

Bits	Name	Type	Function
[26:24]	POSTDIV1	RW	First post-divide value. Post-divide value=POSTDIV1. Reset value 0b1.
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction =FRAC/2 ²⁴ . Reset value 0x0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.36 CLUS_PLL_CTRL0 Register

The CLUS_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register CLUS_PLL_CTRL1, control the settings of clock control PLL CLUSPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CLUS_PLL_CTRL0 Register bit assignments.

Table 5-67: CLUS_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 0b1: 0b0: PLL disabled. 0b1: PLL enable. Reset value 0x1.
[30:26]	-	-	Reserved.

Bits	Name	Type	Function
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0b1.
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x20 , division value=32.
[7:1]	-	-	Reserved.
[0]	HARD_BYPASS		Bypasses PLL to drive input clock directly into the SoC: 0b0: PLL not bypassed.. 0b1: PLL bypassed. Reset value 0b0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.37 CLUS_PLL_CTRL1 Register

The CLUS_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register CLUS_PLL_CTRL0, control the settings of clock control PLL CLUSPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CLUS_PLL_CTRL1 Register bit assignments.

Table 5-68: CLUS_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	Second post-divide value. Post-divide value=POSTDIV2. Reset value 0b1.

Bits	Name	Type	Function
[27]	-	-	Reserved.
[26:24]	POSTDIV1	RW	First post-divide value. Post-divide value=POSTDIV1. Reset value 0b1.
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction =FRAC/2 ²⁴ . Reset value 0x0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.38 SYS_PLL_CTRL0 Register

The SYS_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register SYS_PLL_CTRL1, control the settings of clock control PLL SYSPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SYS_PLL_CTRL0 Register bit assignments.

Table 5-69: SYS_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 0b1: 0b0: PLL disabled. 0b1: PLL enabled Reset value 0x1.
[30:26]	-	-	Reserved.

Bits	Name	Type	Function
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0b1.
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x30, division value=48.
[7:1]	-	-	Reserved.
[0]	HARD_BYPASS		Bypasses PLL to drive input clock directly into the SoC: 0b0: PLL not bypassed. 0b1: PLL bypassed. Reset value 0b0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.39 SYS_PLL_CTRL1 Register

The SYS_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register SYS_PLL_CTRL0, control the settings of clock control PLL SYSSPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SYS_PLL_CTRL1 Register bit assignments.

Table 5-70: SYS_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	Second post-divide value. Post-divide value=POSTDIV2. Reset value 0b1.
[27]	-	-	Reserved.

Bits	Name	Type	Function
[26:24]	POSTDIV1	RW	First post-divide value. Post-divide value=POSTDIV1. Reset value 0b1.
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction $\text{FRAC}/2^{24}$. Reset value 0x0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.40 DMC_PLL_CTRL0 Register

The DMC_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register DMC_PLL_CTRL1, control the settings of clock control PLL DMCPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DMC_PLL_CTRL0 Register bit assignments.

Table 5-71: DMC_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 0b1: 0b0: PLL disabled. 0b1: PLL enabled . Reset value 0x1.
[30:26]	-	-	Reserved.
[25:20]	REFDIV	RW	PLL reference, input clock divider value. Reset value 0b1.

Bits	Name	Type	Function
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x20 , division value=32.
[7:1]	-	-	Reserved.
[0]	HARD_BYPASS	RW	Bypasses PLL to drive input clock directly into the SoC: 0b0: PLL not bypassed. 0b1: PLL bypassed. Reset value 0b0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.41 DMC_PLL_CTRL1 Register

The DMC_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register DMC_PLL_CTRL0, control the settings of clock control PLL DMCPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DMC_PLL_CTRL1 Register bit assignments.

Table 5-72: DMC_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	Second post-divide value. Post-divide value=POSTDIV2. Reset value 0b1.
[27]	-	-	Reserved.

Bits	Name	Type	Function
[26:24]	POSTDIV1	RW	First post-divide value. Post-divide value=POSTDIV1. Reset value 0b1.
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction =FRAC/2 ²⁴ . Reset value 0x0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.42 INT_PLL_CTRL0 Register

The INT_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register INT_PLL_CTRL1, control the settings of clock control PLL INTPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the INT_PLL_CTRL0 Register bit assignments.

Table 5-73: INT_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 0b1: 0b0: PLL disabled. 0b1: PLL enabled . Reset value 0b0.
[30:26]	-	-	Reserved.
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0b1.

Bits	Name	Type	Function
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x20 , division value = 32.
[7:1]	-	-	Reserved.
[0]	HARD_BYPASS	-	Bypasses PLL to drive input clock directly into the SoC: 0b0: PLL not bypassed. 0b1: PLL bypassed. Reset value 0b0.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.43 INT_PLL_CTRL1 Register

The INT_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register INT_PLL_CTRL0, control the settings of clock control PLL INTPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the INT_PLL_CTRL1 Register bit assignments.

Table 5-74: INT_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	Second post-divide value. Post-divide value=POSTDIV2. Reset value 0b1.
[27]	-	-	Reserved.

Bits	Name	Type	Function
[26:24]	POSTDIV1	RW	First post-divide value. Post-divide value=POSTDIV1. Reset value 0b1.
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction = $FRAC/2^{24}$



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.44 GPU_PLL_CTRL1 Register

The GPU_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register GPU_PLL_CTRL0, control the settings of clock control PLL GPUPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the GPU_PLL_CTRL1 Register bit assignments.

Table 5-75: GPU_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved
[30:28]	POSTDIV2	RW	Second post-divide value. Post-divide value=POSTDIV2. Reset value 0x1.
[27]	-	-	Reserved.
[26:24]	POSTDIV1	RW	First post-divide value. Post-divide value=POSTDIV1. Reset value 0x2.

Bits	Name	Type	Function
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction = $\text{FRAC} / 2^{24}$.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.45 GPU_PLL_CTRL0 Register

The GPU_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register GPU_PLL_CTRL1, control the settings of clock control PLL GPUPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the GPU_PLL_CTRL0 Register bit assignments.

Table 5-76: GPU_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 1: 0: PLL disabled. 1: PLL enabled. Reset value 0x0.
[30:26]	-	-	Reserved.
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0x1.
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x1A, division value=26.
[7:1]	-	-	Reserved.

Bits	Name	Type	Function
[0]	HARD_BYPASS		<p>Bypasses PLL to drive input clock directly into the SoC:</p> <p>0: PLL not bypassed.</p> <p>1: PLL bypassed.</p> <p>Reset value 0x0.</p>



Note

The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.46 DPU_PLL_CTRL1 Register

The DPU_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register DPU_PLL_CTRL0, control the settings of clock control PLL DPUPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DPU_PLL_CTRL1 Register bit assignments.

Table 5-77: DPU_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	<p>Second post-divide value.</p> <p>Post-divide value=POSTDIV2.</p> <p>Reset value 0x1.</p>
[27]	-	-	Reserved.
[26:24]	POSTDIV1	RW	<p>First post-divide value.</p> <p>Post-divide value=POSTDIV1.</p> <p>Reset value 0x5.</p>

Bits	Name	Type	Function
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction = $\text{FRAC}/2^{24}$.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.47 DPU_PLL_CTRL0 Register

The DPU_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register DPU_PLL_CTRL1, control the settings of clock control PLL DPUPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DPU_PLL_CTRL0 Register bit assignments.

Table 5-78: DPU_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 1: 0: PLL disabled. 1: PLL enabled. Reset value 0x0.
[30:26]	-	-	Reserved.
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0x2.
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x49 , division value=73.
[7:1]	-	-	Reserved.

Bits	Name	Type	Function
[0]	HARD_BYPASS		<p>Bypasses PLL to drive input clock directly into the SoC:</p> <p>0: PLL not bypassed.</p> <p>1: PLL bypassed.</p> <p>Reset value 0x0.</p>



Note

The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.48 PXL_PLL_CTRL1 Register

The PXL_PLL_CTRL1 Register characteristics are:

Purpose

This register, and register PXL_PLL_CTRL0, control the settings of clock control PLL PXLPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PXL_PLL_CTRL1 Register bit assignments.

Table 5-79: PXL_PLL_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31]	-	-	Reserved.
[30:28]	POSTDIV2	RW	<p>Second post-divide value.</p> <p>Post-divide value=POSTDIV2.</p> <p>Reset value 0x1.</p>
[27]	-	-	Reserved.
[26:24]	POSTDIV1	RW	<p>First post-divide value.</p> <p>Post-divide value=POSTDIV1.</p> <p>Reset value 0x5.</p>

Bits	Name	Type	Function
[23:0]	FRAC	RW	Fractional part of feedback divide value. Fraction = $FRAC/2^{24}$.



The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.49 PXL_PLL_CTRL0 Register

The PXL_PLL_CTRL0 Register characteristics are:

Purpose

This register, and register PXL_PLL_CTRL1, control the settings of clock control PLL PXLPLL.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PXL_PLL_CTRL0 Register bit assignments.

Table 5-80: PXL_PLL_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31]	PLLEN	RW	PLL global enable. After SoC bootup, the PLL is disabled until this bit is set to 1: 0: PLL disabled. 1: PLL enabled. Reset value 0x0.
[30:26]	-	-	Reserved.
[25:20]	REFDIV	RW	PLL reference, input clock, divider value. Reset value 0x5.
[19:8]	FBDIV	RW	PLL feedback divider value. Reset value 0x51 , division value=81.
[7:1]	-	-	Reserved.

Bits	Name	Type	Function
[0]	HARD_BYPASS		<p>Bypasses PLL to drive input clock directly into the SoC:</p> <p>0: PLL not bypassed.</p> <p>1: PLL bypassed.</p> <p>Reset value 0x0.</p>



Note

The example values in this register, and the clock frequency they generate, are part of a clock configuration which enables correct operation of the Morello SoC. Further SoC testing and measurement, by Arm or by other developers, might result in new register values.

5.5.50 SYS_MAN_RESET Register

The SYS_MAN_RESET Register characteristics are:

Purpose

Controls the manual resets of the internal resets at SoC level.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SYS_MAN_RESET Register bit assignments.

Table 5-81: SYS_MAN_RESET Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11]	FORCE_CCIX_APB_RST	RW	<p>CCIX APB reset, CCIX top reset:</p> <p>0b0: Not reset.</p> <p>0b1: Reset.</p> <p>Reset value 0b1.</p>
[10]	FORCE_PCIE_APB_RST	-	<p>PCIe APB reset, PCIe top reset:</p> <p>0b0: Not reset.</p> <p>0b1: Reset.</p> <p>Reset value 0b1.</p>

Bits	Name	Type	Function
[9:8]	-	-	Reserved.
[7]	FORCE_MCP_QSPI_RST	-	MCPQSPICLK manual reset: 0b0: Not reset. 0b1: Reset. Reset value 0b0.
[6]	FORCE_MCP_I2C_RST	-	MCP I2CCLK clock manual reset: 0b0: Not reset. 0b1: Reset. Reset value 0b0.
[5]	FORCE_S_CP_SENSOR_RST	-	SENSORCLK manual reset: 0b0: Not reset. 0b1: Reset. Reset value 0b0.
[4]	FORCE_SCP_QSPI_RST	-	SCPQSPICLK manual reset: 0b0: Not reset. 0b1: Reset. Reset value 0b0.
[3]	FORCE_SCP_I2C_RST	-	SCP I2CCLK manual reset: 0b0: Not reset. 0b1: Reset. Reset value 0b0.
[2]	FORCE_IO_FPGA_TSIF_RST	-	TSIF2XCLK manual reset: 0b0: Not reset. 0b1: Reset. Reset value 0b0.
[1]	FORCE_IO_FPGA_TMIF_RST	-	TMIF2XCLK manual reset: 0b0: Not reset. 0b1: Reset. Reset value 0b0.

Bits	Name	Type	Function
[0]	FORCE_SYS_APB_RST	-	<p>SYSAPBCLK manual reset:</p> <p>0b0: Not reset.</p> <p>0b1: Reset.</p> <p>Reset value 0b0.</p>

5.5.51 BOOT_CTL Register

The BOOT_CTL Register characteristics are:

Purpose

Controls powerup reset hold and MSCP bootup type.

Usage constraints

This register is read-only from APB interface and read/write from serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_CTL Register bit assignments.

Table 5-82: BOOT_CTL Register bit assignments

Bits	Name	Type	Function
[31]	PORESET_HOLD	<p>RO from APB interface.</p> <p>RW from serial interface.</p>	<p>Powerup reset hold:</p> <p>0b0: Do not hold powerup reset.</p> <p>0b1: Hold poweru p reset.</p> <p>Reset value 0b0.</p> <p>This bit is valid only when MSCP_BOOT_TYPE=1 .</p>
[30:1]	-		Reserved.
[0]	MSCP_BOOT_TYPE	<p>RO from APB interface.</p> <p>RW from serial interface.</p>	<p>MSCP bootup type:</p> <p>0b0: Boot from QSPI.</p> <p>0b1: Boot from TLX manager interface.</p> <p>Reset value 0b0.</p>

5.5.52 BOOT_CTRL_STA Register

The BOOT_CTRL_STA Register characteristics are:

Purpose

Stores bootup statuses.

Usage constraints

Bits[31:24] are read/write. Bits[6:0] are read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_CTRL_STA Register bit assignments.

Table 5-83: BOOT_CTRL_STA Register bit assignments

Bits	Name	Type	Function
[31:24]	MSCP_BOOT_STATUS	RW	SCP can use this field to store MSCP bootup status for MCC to read. Reset value 0x00.
[23:7]	-	-	Reserved.
[6]	MCP_ACG_QDENY	RO	MCP ACG QDENY _n . Reset value 0b0.
[5]	MCP_ACG_QACCEPT	RO	MCP ACG QACCEPT _n . Reset value 0b0.
[4]	MCP_QACTIVE	RO	MCP ACG QACTIVE . Reset value 0b0.
[3]	-	RO	Reserved.
[2]	SCP_ACG_QDENY	RO	SCP ACG QDENY _n .
[1]	SCP_QACCEPT	RO	SCP ACG QACCEPT _n . Reset value 0b0.
[0]	SCP_ACG_QACTIVE	RO	SCP ACG QACTIVE . Reset value 0b0.

5.5.53 SCP_BOOT_ADR Register

The SCP_BOOT_ADR Register characteristics are:

Purpose

Stores the SCP bootup address.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCP_BOOT_ADR Register bit assignments.

Table 5-84: SCP_BOOT_ADR Register bit assignments

Bits	Name	Type	Function
[31:0]	ADDRESS	RO from APB interface. RW from serial interface.	Bootup address of SCP. Reset value 0x00000000.

5.5.54 MCP_BOOT_ADR Register

The MCP_BOOT_ADR Register characteristics are:

Purpose

Stores the MCP bootup address.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCP_BOOT_ADR Register bit assignments.

Table 5-85: MCP_BOOT_ADR Register bit assignments

Bits	Name	Type	Function
[31:0]	ADDRESS	RO from APB interface. RW from serial interface.	Bootup address of MCP. Reset value 0x00000000.

5.5.55 PLATFORM_CTRL Register

The PLATFORM_CTRL Register characteristics are:

Purpose

Morello SoC platform control.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PLATFORM_CTRL Register bit assignments.

Table 5-86: PLATFORM_CTRL Register bit assignments

Bits	Name	Type	Function
[31:9]	-	-	Reserved.
[8]	MULTI_CHIP_MODE	RO from APB interface. RW from serial interface.	Multi-chip tie-off value: 0b0: Single chip. 0b1: Multi-chip. Reset value 0b0.
[7:6]	-	-	Reserved.
[5:0]	CHIPID	RO from APB interface. RW from serial interface.	CHIP ID tie-off value in multichip mode. This value is 0b00000 for single chip mode.

5.5.56 TARGETIDAPP Register

The TARGETIDAPP Register characteristics are:

Purpose

CoreSight target ID of Application Processor (AP).

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the TARGETIDAPP Register bit assignments.

Table 5-87: TARGETIDAPP Register bit assignments

Bits	Name	Type	Function
[31:0]	ID	RO from APB interface. RW from serial interface.	CoreSight target ID of AP. Reset value 0x07B30477.

5.5.57 TARGETIDSCP Register

The TARGETIDSCP Register characteristics are:

Purpose

Stores the SCP bootup address.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCP_BOOT_ADR Register bit assignments.

Table 5-88: SCP_BOOT_ADR Register bit assignments

Bits	Name	Type	Function
[31:0]	ID	RO from APB interface. RW from serial interface.	CoreSight target ID of SCP. Reset value 0x07B40477.

5.5.58 TARGETIDMCP Register

The TARGETIDMCP Register characteristics are:

Purpose

Stores the MCP bootup address.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the TARGETIDMCP Register bit assignments.

Table 5-89: TARGETIDMCP Register bit assignments

Bits	Name	Type	Function
[31:0]	ID	RO from APB interface. RW from serial interface.	CoreSight target ID of MCP. Reset value 0x07B50477.

5.5.59 BOOT_GPR0 Register

The BOOT_GPR0 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR0 Register bit assignments.

Table 5-90: BOOT_GPR0 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface. RW from serial interface.	Bootup general-purpose register 0. Reset value 0x00000000.

5.5.60 BOOT_GPR1 Register

The BOOT_GPR1 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR1 Register bit assignments.

Table 5-91: BOOT_GPR1 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface.	Bootup general-purpose register 1.
		RW from serial interface.	Reset value 0x00000000.

5.5.61 BOOT_GPR2 Register

The BOOT_GPR2 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR2 Register bit assignments.

Table 5-92: BOOT_GPR2 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface.	Bootup general-purpose register 2.
		RW from serial interface.	Reset value 0x00000000.

5.5.62 BOOT_GPR3 Register

The BOOT_GPR3 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR3 Register bit assignments.

Table 5-93: BOOT_GPR3 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface.	Bootup general-purpose register 3.
		RW from serial interface.	Reset value 0x00000000.

5.5.63 BOOT_GPR4 Register

The BOOT_GPR4 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR4 Register bit assignments.

Table 5-94: BOOT_GPR4 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface.	Bootup general-purpose register 4.
		RW from serial interface.	Reset value 0x00000000.

5.5.64 BOOT_GPR5 Register

The BOOT_GPR5 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR5 Register bit assignments.

Table 5-95: BOOT_GPR5 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface. RW from serial interface.	Bootup general-purpose register 5. Reset value 0x00000000.

5.5.65 BOOT_GPR6 Register

The BOOT_GPR6 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR6 Register bit assignments.

Table 5-96: BOOT_GPR6 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface.	Bootup general-purpose register 6.
		RW from serial interface.	Reset value 0x00000000.

5.5.66 BOOT_GPR7 Register

The BOOT_GPR7 Register characteristics are:

Purpose

Bootup general-purpose register. This register enables an external controller to pass bootup configuration information into the Morello SoC before the release of the powerup reset. The register does not export any control from the SCC.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the BOOT_GPR7 Register bit assignments.

Table 5-97: BOOT_GPR7 Register bit assignments

Bits	Name	Type	Function
[31:0]	REG	RO from APB interface.	Bootup general-purpose register 7.
		RW from serial interface.	Reset value 0x00000000.

5.5.67 INSTANCE_ID Register

The INSTANCE_ID Register characteristics are:

Purpose

SWJ-DP instance ID register.

Usage constraints

This register is read-only from the APB interface and read/write from the serial interface.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the INSTANCE_ID Register bit assignments.

Table 5-98: INSTANCE_ID Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3:0]	ID	RO from APB interface. RW from serial interface.	SWJ-DP instance ID register. Reset value 0b0000.

5.5.68 PCIE_BOOT_CTRL Register

The PCIE_BOOT_CTRL Register characteristics are:

Purpose

Enables reset of sticky bits in the PCIe and CCIX controllers during reset of the PCIe and CCIX controllers.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PCIE_BOOT_CTRL Register bit assignments.

Table 5-99: PCIE_BOOT_CTRL Register bit assignments

Bits	Name	Type	Function
[31:2]	-	-	Reserved.
[1]	CCIX_STICKY_RST_EN	RW	Enable reset of all sticky bits in the CCIX controller during CCIX controller reset: 0b0: Not enable reset of sticky bits. 0b1: Enable reset of sticky bits . Reset value 0b1.
[0]	PCIE_STICKY_RST_EN	RW	Enable reset of all sticky bits in the PCIe controller during PCIe controller reset: 0b0: Not enable reset of sticky bits. 0b1: Enable reset of sticky bits . Reset value 0b1.

5.5.69 GPU_CTRL Register

The GPU_CTRL Register characteristics are:

Purpose

GPU control register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the GPU_CTRL Register bit assignments.

Table 5-100: GPU_CTRL Register bit assignments

Bits	Name	Type	Function
[31:5]	-	-	Reserved.
[4]	GPU_NIDEN	RW	Non-invasive debug enable. Reset value 0x1.
[3]	GPU_DBGEN	RW	Debug enable. Reset value 0x1.
[2:0]	GPU_STRIPING_ GRANULE	RW	4KB Norr L2 cache granule size. Reset value 0x0.

5.5.70 DBG_AUTHN_CTRL Register

The DBG_AUTHN_CTRL Register characteristics are:

Purpose

Drives the CoreSight™ authentication external interface.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DBG_AUTHN_CTRL Register bit assignments.

Table 5-101: DBG_AUTHN_CTRL Register bit assignments

Bits	Name	Type	Function
[31:3]	-	-	Reserved.
[2]	DBG_SPNIDEN	RW	Secure non-invasive debug enable: 0b0: Disable. 0b1: Enable. Reset value 0b1.
[1]	DBG_SPIDEN	RW	Secure invasive debug enable: 0b0: Disable. 0b1: Enable. Reset value 0b1.
[0]	DBG_DEVICEEN	RW	Global external debug enable: 0b0: Disable. 0b1: Enable. Reset value 0b1.

5.5.71 CTIO_CTRL Register

The CTIO_CTRL Register characteristics are:

Purpose

CTI trigger mask register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CTIO_CTRL Register bit assignments.

Table 5-102: CTIO_CTRL Register bit assignments

Bits	Name	Type	Function
[31:16]	-	-	Reserved.
[15:8]	TODBGENSEL	RW	CTI TODBGENSEL input. Reset value 0x00 .

Bits	Name	Type	Function
[7:0]	TINIDENSEL	RW	CTI TINIDENSEL input. Reset value 0x0.

5.5.72 CTI1_CTRL Register

The CTI1_CTRL Register characteristics are:

Purpose

CTI trigger mask register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CTI1_CTRL Register bit assignments.

Table 5-103: CTI1_CTRL Register bit assignments

Bits	Name	Type	Function
[31:16]	-	-	Reserved.
[15:8]	TODBGENSEL	RW	CTI TODBGENSEL input. Reset value 0x00 .
[7:0]	TINIDENSEL	RW	CTI TINIDENSEL input. Reset value 0x0.

5.5.73 CTI0TO3_CTRL Register

The CTI0TO3_CTRL Register characteristics are:

Purpose

CTI trigger mask register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CTI0TO3_CTRL Register bit assignments.

Table 5-104: CTI0TO3_CTRL Register bit assignments

Bits	Name	Type	Function
[31:16]	-	-	Reserved.
[15:8]	TODBGENSEL	RW	CTI TODBGENSEL input. Reset value 0x00 .
[7:0]	TINIDENSEL	RW	CTI TINIDENSEL input. Reset value 0x00 .

5.5.74 MCP_WDOGCTI_CTRL Register

The MCP_WDOGCTI_CTRL Register characteristics are:

Purpose

CTI trigger mask register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the MCP_WDOGCTI_CTRL Register bit assignments.

Table 5-105: MCP_WDOGCTI_CTRL Register bit assignments

Bits	Name	Type	Function
[31:16]	-	-	Reserved.
[15:8]	TODBGENSEL	RW	CTI TODBGENSEL input. Reset value 0x00 .
[7:0]	-	-	Reserved.

5.5.75 SCP_WDOGCTI_CTRL Register

The SCP_WDOGCTI_CTRL Register characteristics are:

Purpose

CTI trigger mask register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCP_WDOGCTI_CTRL Register bit assignments.

Table 5-106: SCP_WDOGCTI_CTRL Register bit assignments

Bits	Name	Type	Function
[31:16]	-	-	Reserved.
[15:8]	TODBGENSEL	RW	CTI TODBGENSEL input. Reset value 0x00.
[7:0]	-	-	Reserved.

5.5.76 DBGEXPCTI_CTRL Register

The DBGEXPCTI_CTRL Register characteristics are:

Purpose

CTI trigger mask register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DBGEXPCTI_CTRL Register bit assignments.

Table 5-107: DBGEXPCTI_CTRL Register bit assignments

Bits	Name	Type	Function
[31:24]	TODBGENSEL2	RW	CTI2 TODBGENSEL input. Reset value 0x00 .
[23:16]	TINIDENSEL2	RW	CTI2 TINIDENSEL input. Reset value 0x00 .
[15:8]	TODBGENSEL1	RW	CTI1 TODBGENSEL input. Reset value 0x00 .
[7:0]	TINIDENSEL1	RW	CTI1 TINIDENSEL input. Reset value 0x00 .

5.5.77 PCIE_PM_CTRL Register

The PCIE_PM_CTRL Register characteristics are:

Purpose

PCIe power control register.

Usage constraints

Bit[1] is read-only. Bit[0] is read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PCIE_PM_CTRL Register bit assignments.

Table 5-108: PCIE_PM_CTRL Register bit assignments

Bits	Name	Type	Function
[31:2]	-	-	Reserved.
[1]	PM_ACK	RO	PCIe powerup acknowledgement: 0b0: Not acknowledge. 0b1: Acknowledge. Reset value 0b0.
[0]	PM_REQ	RW	PCIe powerup request: 0b0: No effect. 0b1: Request powerup. Reset value 0b0.

5.5.78 CCIX_PM_CTRL Register

The CCIX_PM_CTRL Register characteristics are:

Purpose

CCIX power control register.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CCIX_PM_CTRL Register bit assignments.

Table 5-109: CCIX_PM_CTRL Register bit assignments

Bits	Name	Type	Function
[31:2]	-	-	Reserved.
[1]	PM_ACK	RO	CCIX powerup acknowledgement: 0b0: Not acknowledge. 0b1: Acknowledge. Reset value 0b0.
[0]	PM_REQ	RW	CCIX powerup request: 0b0: No effect. 0b1: Request powerup. Reset value 0b0.

5.5.79 SCDBG_CTRL Register

The SCDBG_CTRL Register characteristics are:

Purpose

SCC scan-based debug control register.

Usage constraints

Bits[9:8] and bits[5:4] are read-only. Bits[15:0] are reserved. All other bits are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the SCDBG_CTRL Register bit assignments.

Table 5-110: SCDBG_CTRL Register bit assignments

Bits	Name	Type	Function
[31:16]	MANUAL_TRIG_DELAY	RW	Number of REFCLK cycles to wait after a write to MANUAL_TRIGGER register. Default 0x0000. Maximum 0xFFFF. Reset value 0x0000.
[15:10]	-	-	Reserved.
[9]	SOC_ELAOUTUT0	RO	Or-ed SoC ELA EALOUTPUT[0]. Reset value 0b0.
[8]	MODE_STATUS	RO	Sticky signal which indicates that the Morello SoC has entered Scan-based debug mode: 0b0: Not Scan-based debug mode. 0b1: Scan-based debug mode. Reset value 0b0.
[7]	MANUAL_TRIG	RW	Triggers scan-based dump if TRIG_MANUAL is enabled: 0b0: No effect. 0b1: Trigger scan-based dump. Reset value 0b0.
[6]	TRIG_MANUAL	RW	Include manual trigger: 0b0: No effect. 0b1: Include manual trigger. Reset value 0b0.
[5]	TRIG_ELA_SOC	RO	Or-ed Logic Analyzer, ELA, STOPCLOCK trigger from all Morello SoC ELAs. Reset value 0b0.
[4]	TRIG_ELA_AP	RO	Or-ed Logic Analyzer, ELA, STOPCLOCK trigger from both clusters. Reset value 0b0.
[3]	TRIG_CTHALT_C1	RW	Include cluster 1 cross trigger halt event, OR function of all PE cross trigger halt events. Reset value 0b0.
[2]	TRIG_CTHALT_C0	RW	Include cluster 0 cross trigger halt event, OR function of all PE cross trigger halt events. Reset value 0b0.
[1]	TRIG_SS_RESETRREQ	RW	Include Manageability Control Processor (MC) and System Control Processor (SCP) subsystem reset request. Reset value 0b0.

Bits	Name	Type	Function
[0]	MASTER_EN	RW	<p>Scan-based debug manager enable. This bit must be 0b1 to enter SCD mode.</p> <p>0b0: Not enable.</p> <p>0b1: Enable.</p> <p>Reset value 0b0.</p>

5.5.80 EXP_IF_CTRL Register

The EXP_IF_CTRL Register characteristics are:

Purpose

Controls certain CCIX and PCIe activity.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the EXP_IF_CTRL Register bit assignments.

Table 5-111: EXP_IF_CTRL Register bit assignments

Bits	Name	Type	Function
[31:24]	-	-	Reserved.
[23:16]	TSIF_WIN_ADDR	RW	<p>Controls the location of the TSIF 1TB address window inside the Application Processor memory map:</p> <p>0-4TB for single chip system.</p> <p>0-8TB for two chip system.</p> <p>This field should be set before the first transaction from any TSIF masters and should not be changed afterwards.</p> <p>Reset value 0x0000_0000.</p>
[15:2]	-	-	Reserved.

Bits	Name	Type	Function
[1]	ROUNDROBIN_TBU_CCIX	RW	<p>Defines the Micro TLB entry replacement policy for the PCIe AXI expansion interface.</p> <p>0b0: The Micro TLB uses a pseudo Least Recently Used (LRU) replacement policy. This policy typically provides the best average performance. However, when multiple translations are prefetched using a StashTranslation transaction, they might evict each other.</p> <p>0b1: The Micro TLB uses a round-robin replacement policy. This policy enables prefetch multiple translations using a StashTranslation transaction without evictions if the Micro TLB size is not exceeded.</p> <p>To avoid evictions, set this bit to 0b1 if a real-time upstream manager prefetches translations.</p>
[0]	ROUNDROBIN_TBU_PCIE	RW	<p>Defines the Micro TLB entry replacement policy for the CCIX AXI expansion interface.</p> <p>0b0: The Micro TLB uses a pseudo Least Recently Used (LRU) replacement policy. This policy typically provides the best average performance. However, when multiple translations are prefetched using a StashTranslation transaction, they might evict each other.</p> <p>0b1: The Micro TLB uses a round-robin replacement policy. This policy enables prefetch multiple translations using a StashTranslation transaction without evictions if the Micro TLB size is not exceeded.</p> <p>To avoid evictions, set this bit to 0b1 if a real-time upstream manager prefetches translations.</p>

5.5.81 RO_CTRL Register

The RO_CTRL Register characteristics are:

Purpose

Enables ring oscillator to directly measure silicon liveliness.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the RO_CTRL Register bit assignments.

Table 5-112: RO_CTRL Register bit assignments

Bits	Name	Type	Function
[31:1]	-	-	Reserved.

Bits	Name	Type	Function
[0]	RO_EN	RW	<p>Enables and disables ring oscillator:</p> <p>0b0: Disable ring oscillator.</p> <p>0b1: Enable ring oscillator.</p> <p>Reset value 0b1.</p>

5.5.82 CMN_CCIX_CTRL Register

The CMN_CCIX_CTRL Register characteristics are:

Purpose

CCIX control register.

Usage constraints

Bits[27:2] and bits[19:17] are read-only. Bit[24] and bits[16:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CMN_CCIX_CTRL Register bit assignments.

Table 5-113: CMN_CCIX_CTRL Register bit assignments

Bits	Name	Type	Function
[31:28]	-	-	Reserved.
[27]	CXLA_CXCLK_QDENY	RO	<p>QDENY of CXLA CXCLK control Q channel at CXS interface side.</p> <p>Reset value 0b0.</p>
[26]	CXLA_CXCLK_QACCEPT	RO	<p>QACCEPTn of CXLA CXCLK control Q channel at CXS interface side.</p> <p>Reset value 0b0.</p>
[25]	CXLA_CXCLK_QACTIVE	RO	<p>QACTIVE of CXLA CXCLK control Q channel at CXS interface side.</p> <p>Reset value 0b0.</p>
[24]	CXLA_CXCLK_QREQ	RW	<p>QREQn of CXLA CXCLK control Q channel at CXS interface side.</p> <p>This bit maintains its reset value while the CCIX subsystem is operating. It is only used to complete Q-channel clock down handshake when the CCIX subsystem, CCIX PCIe controller, needs reset while the main part of CMN-600 is running. This is useful for CCIX subsystem error clearance.</p> <p>Reset value 0b1.</p>
[23:20]	-	-	Reserved.
[19]	CXLA_PWR_QDENY	RO	QDENY of CXLA power control Q channel at CXS interface side.

Bits	Name	Type	Function
[18]	CXLA_PWR_QACCEPT	RO	QACCEPTN of CXLA power control Q channel at CXS interface side.
[17]	CXLA_PWR_QACTIVE	RO	QACTIVE of CXLA power control Q channel at CXS interface side.
[16]	CXLA_PWR_QREQ	RW	<p>QREQn of CXLA power control Q channel at CXS interface side.</p> <p>This bit maintains its reset value while the CCIX subsystem is operating. It is only used to complete Q-channel power down handshakes when the CCIX subsystem, CCIX PCIe controller, needs reset while the main part of CMN-600 is running. This is useful for CCIX subsystem error clearance.</p> <p>Reset value 0b1.</p>
[15:0]	PCIE_BUS_NUM	RW	<p>The PCIe ID{BUS_NUM[15:8], DEVICE_NUM[7:3], FUNCTION_NUM[2:0]} used for CMN-600 to form its PCIe header.</p> <p>When the CCIX is configured as RP, this field must be set to 0x0.</p> <p>When the CCIX is configured as EP, the SCP reads the End Point Bus and Device Number Register of CCIX enabled PCIe controller and set the value accordingly.</p> <p>Interrupt ccix_bus_device_change_irq indicates a change of value in the controller.</p>

5.5.83 STM_CTRL Register

The STM_CTRL Register characteristics are:

Purpose

Non-secure guaranteed access control.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the STM_CTRL Register bit assignments.

Table 5-114: STM_CTRL Register bit assignments

Bits	Name	Type	Function
[31:1]	-	-	Reserved.

Bits	Name	Type	Function
[0]	NSGUAREN	RW	<p>The top level static configuration port, NSGUAREN port, NSGUAREN, controls the behavior of the the <i>System Trace Macrocell</i> (STM) for Non-secure guaranteed AXI accesses:</p> <p>0b0: Non-secure guaranteed accesses behave like invariant timing accesses, that is, the AXI does not stall.</p> <p>0b1: Non-secure guaranteed accesses are enabled, that is, the AXI can stall and the trace output is guaranteed.</p> <p>Reset value 0b0.</p>

5.5.84 AXI_OVRD_PCIE Register

The AXI_OVRD_PCIE Register characteristics are:

Purpose

Controls PCIe AXI subordinate expansion interface override.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the AXI_OVRD_PCIE Register bit assignments.

Table 5-115: AXI_OVRD_PCIE Register bit assignments

Bits	Name	Type	Function
[31:22]	-	-	Reserved.
[21:20]	AWDOMAIN_TPH	RW	<p>Override value of AWCACHE when TPH values are present.</p> <p>Reset value 0b11.</p>
[19:16]	AWCACHE_TPH	RW	<p>Override value of AWCACHE when TPH values are present.</p> <p>Reset value 0b0000.</p>
[15:14]	-	-	Reserved.
[13:12]	ARDOMAIN	RW	<p>Override value of ARCACHE.</p> <p>Reset value 0b11.</p>
[11:8]	ARCACHE	RW	<p>Override value of AWCACHE.</p> <p>Reset value 0b0000.</p>
[7:6]	-	-	Reserved.
[5:4]	AWDOMAIN	RW	<p>Override value of AWCACHE when TPH values are not present.</p> <p>Reset value 0b11.</p>

Bits	Name	Type	Function
[3:0]	AWCACHE	RW	Override value of AWCACHE when TPH values are not present. Reset value 0b0000.

5.5.85 AXI_OVRD_CCIX Register

The AXI_OVRD_CCIX Register characteristics are:

Purpose

Controls CCIX AXI subordinate expansion interface override.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the AXI_OVRD_CCIX Register bit assignments.

Table 5-116: AXI_OVRD_CCIX Register bit assignments

Bits	Name	Type	Function
[31:22]	-	-	Reserved.
[21:20]	AWDOMAIN_TPH	RW	Override value of AWCACHE when TPH values are present. Reset value 0b11 .
[19:16]	AWCACHE_TPH	RW	Override value of AWCACHE when TPH values are present. Reset value 0b0000.
[15:14]	-	-	Reserved.
[13:12]	ARDOMAIN	RW	Override value of ARCACHE. Reset value 0b11 .
[11:8]	ARCACHE	RW	Override value of AWCACHE. Reset value 0b0000.
[7:6]	-	-	Reserved.
[5:4]	AWDOMAIN	RW	Override value of AWCACHE when TPH values are not present. Reset value 0b11 .
[3:0]	AWCACHE	RW	Override value of AWCACHE when TPH values are not present. Reset value 0b0000.

5.5.86 AXI_OVRD_TSIF Register

The AXI_OVRD_TSIF Register characteristics are:

Purpose

Controls TSIF AXI subordinate expansion interface override.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the AXI_OVRD_TSIF Register bit assignments.

Table 5-117: AXI_OVRD_TSIF Register bit assignments

Bits	Name	Type	Function
[31:14]	-	-	Reserved.
[13:12]	ARDOMAIN	RW	Override value of ARCACHE. Reset value 0b11 .
[11:6]	-	-	Reserved.
[5:4]	AWDOMAIN	RW	Override value of AWCACHE. Reset value 0b11 .
[3:0]	-	-	Reserved.

5.5.87 GPU_TEXFMTENABLE Register

The GPU_TEXFMTENABLE Register characteristics are:

Purpose

Controls GPU compressed texture format support.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the GPU_TEXFMTENABLE Register bit assignments.

Table 5-118: GPU_TEXFMTENABLE Register bit assignments

Bits	Name	Type	Function
[31:0]	NORR_TEXFMTENABLE	RW	Norr compressed texture format support.

5.5.88 TRACE_PAD_CTRL0 Register

The TRACE_PAD_CTRL0 Register characteristics are:

Purpose

Controls the drive strengths and slew rates of trace data output pads.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the TRACE_PAD_CTRL0 Register bit assignments.

Table 5-119: TRACE_PAD_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31:29]	-	-	Reserved.
[28]	IO_SR_TRACE_DATA 3	RW	Slew rate control of trace port output pads TRACE_DATA[31:24]: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[27:26]	-	-	Reserved.
[25:24]	IO_DS_TRACE_DATA 3	RW	Drive strength control of trace port output pads TRACE_DATA[31:24]: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.
[23:21]	-	-	Reserved.

Bits	Name	Type	Function
[20]	IO_SR_TRACE_DATA 2	RW	Slew rate control of trace port output pads TRACE_DATA[23:16]: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[19:18]	-	-	Reserved.
[17:16]	IO_DS_TRACE_DATA 2	RW	Drive strength control of trace port output pads TRACE_DATA[23:16]: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.
[15:13]	-	-	Reserved.
[12]	IO_SR_TRACE_DATA 1	RW	Slew rate control of trace port output pads TRACE_DATA[15:8]: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[11:10]	-	-	Reserved.
[9:8]	IO_DS_TRACE_DATA 1	RW	Drive strength control of trace port output pads TRACE_DATA[15:8]: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.
[7:5]	-	-	Reserved.
[4]	IO_SR_TRACE_DATA 0	RW	Slew rate control of trace port output pads TRACE_DATA[7:0]: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[3:2]	-	-	Reserved.

Bits	Name	Type	Function
[1:0]	IO_DS_TRACE_DATA 0	RW	Drive strength control of trace port output pads TRACE_DATA[7:0]: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.

5.5.89 TRACE_PAD_CTRL1 Register

The TRACE_PAD_CTRL1 Register characteristics are:

Purpose

Controls the drive strengths and slew rates of the trace clock output pads.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the TRACE_PAD_CTRL1 Register bit assignments.

Table 5-120: TRACE_PAD_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31:13]	-	-	Reserved.
[12]	IO_SR_TRACE_CLK_B	RW	Slew rate control of trace port output pad TRACE_CLK_B: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[11:10]	-	-	Reserved.

Bits	Name	Type	Function
[9:8]	IO_DS_TRACE_CLK_B	RW	Drive strength control of trace port output pad TRACE_CLK_B: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01 .
[7:5]	-	-	Reserved.
[4]	IO_SR_TRACE_CLK_A	RW	Slew rate control of trace port output pad TRACE_CLK_A: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[3:2]	-	-	Reserved.
[1:0]	IO_DS_TRACE_CLK_A	RW	Drive strength control of trace port output pad TRACE_CLK_A: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01 .

5.5.90 IOFPGA_TMIF_PAD_CTRL Register

The IOFPGA_TMIF_PAD_CTRL Register characteristics are:

Purpose

Controls the drive strengths and slew rates of IOFPGA AXI TMIF output pads.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the IOFPGA_TMIF_PAD_CTRL Register bit assignments.

Table 5-121: IOFPGA_TMIF_PAD_CTRL Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20]	IO_SR_IOFPGA_AXI_TMIF_CLK	RW	Slew rate control of IOFPGA AXI TMIF output pad IOFPGA_TMIF_CLK_O: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[19:18]	-	-	Reserved.
[17:16]	IO_DS_IOFPGA_AXI_TMIF_CLK	RW	Drive strength control of IOFPGA AXI TMIF output pad IOFPGA_TMIF_CLK_O: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.
[15:13]	-	-	Reserved.
[12]	IO_SR_IOFPGA_AXI_TMIF_CTL	RW	Slew rate control of IOFPGA AXI TMIF output pads IOFPGA_TMIF_VALID_O and IOFPGA_TMIF_CTL_O: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[11:10]	-	-	Reserved.
[9:8]	IO_DS_IOFPGA_AXI_TMIF_CTL	RW	Drive strength control of IOFPGA AXI TMIF output pads IOFPGA_TMIF_VALID_O and IOFPGA_TMIF_CTL_O: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.
[7:5]	- -		Reserved.

Bits	Name	Type	Function
[4]	IO_SR_IOFPGA_AXI_TMIF_DATA	RW	Slew rate control of IOFPGA AXI TMIF output pads IOFPGA_TMIF_DATA_O[7:0]: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[3:2]	-	-	Reserved.
[1:0]	IO_DS_IOFPGA_AXI_TMIF_DATA	RW	Drive strength control of IOFPGA AXI TMIF output pads IOFPGA_TMIF_DATA_O[7:0]: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.

5.5.91 IOFPGA_TSIF_PAD_CTRL Register

The IOFPGA_TSIF_PAD_CTRL Register characteristics are:

Purpose

Controls the drive strengths and slew rates of IOFPGA AXI TSIF output pads.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the IOFPGA_TSIF_PAD_CTRL Register bit assignments.

Table 5-122: IOFPGA_TSIF_PAD_CTRL Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20]	IO_SR_IOFPGA_AXI_TSIF_CLK	RW	Slew rate control of IOFPGA AXI TSIF output pad IOFPGA_TSIF_CLK_O: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[19:18]	-	-	Reserved.

Bits	Name	Type	Function
[17:16]	IO_DS_IOFPGA_AXI_TSIF_CLK	RW	Drive strength control of IOFPGA AXI TSIF output pad IOFPGA_TSIF_CLK_O: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.
[15:13]	-	-	Reserved.
[12]	IO_SR_IOFPGA_AXI_TSIF_CTL	RW	Slew rate control of IOFPGA AXI TSIF output pads IOFPGA_TSIF_VALID_O and IOFPGA_TSIF_CTL_O[1:0]: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[11:10]	-	-	Reserved.
[9:8]	IO_DS_IOFPGA_AXI_TSIF_CTL	RW	Drive strength control of IOFPGA AXI TSIF output pads IOFPGA_TSIF_VALID_O and IOFPGA_TSIF_CTL_O[1:0]: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.
[7:5]	-	-	Reserved.
[4]	IO_SR_IOFPGA_AXI_TSIF_DATA	RW	Slew rate control of IOFPGA AXI TSIF output pads IOFPGA_TSIF_DATA_O[7:0]: 0b0: Fast. 0b1: Slow. Reset value 0b1.
[3:2]	-	-	Reserved.

Bits	Name	Type	Function
[1:0]	IO_DS_IOFPGA_AXI_TSIF_DATA	RW	Drive strength control of IOFPGA AXI TSIF output pads IOFPGA_TSIF_DATA_O[7:0]: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b01.

5.5.92 DISPLAY_PAD_CTRL0 Register

The DISPLAY_PAD_CTRL0 Register characteristics are:

Purpose

Controls the drive strengths and slew rates of the DPU output pads.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DISPLAY_PAD_CTRL0 Register bit assignments.

Table 5-123: DISPLAY_PAD_CTRL0 Register bit assignments

Bits	Name	Type	Function
[31:29]	-	-	Reserved.
[28]	IO_SR_DATAEN	RW	Slew rate control of display output pad DATAEN: 0b0: Fast. 0b1: Slow. Reset value 0b0.
[27:26]	-	-	Reserved.

Bits	Name	Type	Function
[25:24]	IO_DS_DATAEN	RW	Drive strength control of display output pad DATAEN: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b1.
[23:21]	-	-	Reserved.
[20]	IO_SR_VSYNC	RW	Slew rate control of display output pad VSYNC: 0b0: Fast. 0b1: Slow. Reset value 0b0.
[19:18]	-	-	Reserved.
[17:16]	IO_DS_VSYNC	RW	Drive strength control of display output pad VSYNC: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b1.
[15:13]	-	-	Reserved.
[12]	IO_SR_HSYNC	RW	Slew rate control of display output pad HSYNC: 0b0: Fast. 0b1: Slow. Reset value 0b0.
[11:10]	-	-	Reserved.
[9:8]	IO_DS_HSYNC	RW	Drive strength control of display output pad HSYNC: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b1.

Bits	Name	Type	Function
[7:5]	-	-	Reserved.
[4]	IO_SR_PXLCLK	RW	Slew rate control of display output pad PXLCLK: 0b0: Fast. 0b1: Slow. Reset value 0b0.
[3:2]	-	-	Reserved.
[1:0]	IO_DS_PXLCLK	RW	Drive strength control of display output pad PXLCLK: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b1.

5.5.93 DISPLAY_PAD_CTRL1 Register

The DISPLAY_PAD_CTRL1 Register characteristics are:

Purpose

Controls the drive strengths and slew rates of the DPU output pads.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the DISPLAY_PAD_CTRL1 Register bit assignments.

Table 5-124: DISPLAY_PAD_CTRL1 Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20]	IO_SR_PXLDTAB	RW	Slew rate control of display output pads PXLDTAB_0 ~ PXLDTAB_7: 0b0: Fast. 0b1: Slow. Reset value 0b0.

Bits	Name	Type	Function
[19:18]	-	-	Reserved.
[17:16]	IO_DS_PXLDATAB	RW	Drive strength control of display output pads PXLDATAB_0 ~ PXLDATAB_7: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b1.
[15:13]	-	-	Reserved.
[12]	IO_SR_PXLDTAG	RW	Slew rate control of display output pads PXLDTAG_0 ~ PXLDTAG_7: 0b0: Fast. 0b1: Slow. Reset value 0b0.
[11:10]	-	-	Reserved.
[9:8]	IO_DS_PXLDTAG	RW	Drive strength control of display output pads PXLDTAG_0 ~ PXLDTAG_7: 0b00: 2mA. 0b01: 8mA. 0b10: 4mA. 0b11: 12mA. Reset value 0b1.
[7:5]	-	-	Reserved.
[4]	IO_SR_PXLDTAR	RW	Slew rate control of display output pads PXLDTAR_0 ~ PXLDTAR_7: 0b0: Fast. 0b1: Slow. Reset value 0b0.
[3:2]	-	-	Reserved.

Bits	Name	Type	Function
[1:0]	IO_DS_PXLDATAR	RW	<p>Drive strength control of display output pads PXLDATAR_0 ~ PXLDATAR_7:</p> <p>0b00: 2mA.</p> <p>0b01: 8mA.</p> <p>0b10: 4mA.</p> <p>0b11: 12mA.</p> <p>Reset value 0b1.</p>

5.5.94 APB_CTRL_CLR Register

The APB_CTRL_CLR Register characteristics are:

Purpose

Controls reversion to serial control of the register at the specified base address.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the APB_CTRL_CLR Register bit assignments.

Table 5-125: APB_CTRL_CLR Register bit assignments

Bits	Name	Type	Function
[31:12]	NUMBER	RW	Writing A50F5 to this field sets the register, whose base address bits[11:0] specify, to serial control.
[11:0]	BASE_ADDRESS	RW	Base address of register which reverts to serial control when A50F5 is written to bits[31:12].

5.5.95 PID4 Register

The PID4 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PID4 Register bit assignments.

Table 5-126: PID4 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	PID4	RO	Peripheral ID 4 identification. Reset value 0x04 .

5.5.96 PID0 Register

The PID0 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PID0 Register bit assignments.

Table 5-127: PID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	PID0	RO	Peripheral ID 0 identification. Reset value 0xAF .

5.5.97 PID1 Register

The PID1 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PID1 Register bit assignments.

Table 5-128: PID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	PID1	RO	Peripheral ID 1 identification. Reset value 0xB0 .

5.5.98 PID2 Register

The PID2 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PID2 Register bit assignments.

Table 5-129: PID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	PID2	RO	Peripheral ID 2 identification. Reset value 0x0B .

5.5.99 PID3 Register

The PID3 Register characteristics are:

Purpose

Stores peripheral identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the PID3 Register bit assignments.

Table 5-130: PID3 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	PID3	RO	Peripheral ID 3 identification. Reset value 0x00 .

5.5.100 CID0 Register

The CID0 Register characteristics are:

Purpose

Stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CID0 Register bit assignments.

Table 5-131: CID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	CID3	RO	Component ID 3 identification. Reset value 0x0D .

5.5.101 CID1 Register

The CID1 Register characteristics are:

Purpose

Stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CID1 Register bit assignments.

Table 5-132: CID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	CID1	RO	Component ID 1 identification. Reset value 0xF0 .

5.5.102 CID2 Register

The CID2 Register characteristics are:

Purpose

Stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CID2 Register bit assignments.

Table 5-133: CID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	CID2	RO	Component ID 2 identification. Reset value 0x05 .

5.5.103 CID3 Register

The CID3 Register characteristics are:

Purpose

Stores component identification information.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [Serial Configuration Control registers summary](#).

The following table shows the CID3 Register bit assignments.

Table 5-134: CID3 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	CID3	RO	Component ID 3 identification. Reset value 0xB1 .

5.6 APB system registers

The IOFPGA contains the APB system registers.

5.6.1 APB system register summary

The base memory address of the APB system registers in the IOFPGA is 0x1C01_0000.

The following table shows the registers in address offset order from the base memory address.

Base memory address

0x1C01_0000

Table 5-135: Morello SDP APB system register summary

Offset	Name	Type	Reset	Width	Description
0x0000	SYS_ID	RO	0xFFFFFFFF	32	See SYS_ID Register .
0x0004	SYS_SW	RO/RW	0x000000XX	32	See SYS_SW Register .
0x0008	SYS_LED	RO/RW	0x000000XX	32	See SYS_LED Register .
0x0024	SYS_100HZ	RO/RW	0xFFFFFFFF	32	See SYS_100HZ Register .
0x0030	SYS_FLAG	RO	0x00000000	32	See SYS_FLAG Registers .

Offset	Name	Type	Reset	Width	Description
0x0030	SYS_FLAGSSET	WO	-	32	See SYS_FLAG Registers .
0x0034	SYS_FLAGSCLR	WO	-	32	See SYS_FLAG Registers .
0x0038	SYS_NVFLAGS	RO	0x00000000	32	See SYS_FLAG Registers .
0x0038	SYS_NVFLAGSSET	WO	-	32	See SYS_FLAG Registers .
0x003C	SYS_NVFLAGSCLR	WO	-	32	See SYS_FLAG Registers .
0x0058	SYS_CFGSW	RO/RW	0x000000XX	32	See SYS_CFGSW Register .
0x005C	SYS_24MHZ	RO	0xFFFFFFFF	32	See SYS_24MHZ Register .
0x0070	SYS_PCIE_CNTL	RW	0x0000000X	32	See SYS_PCIE_CNTL Register .
0x0074	SYS_PCIE_GBE_L	RO	0xFFFFFFFF	32	See SYS_PCIE_GBE Register .
0x0078	SYS_PCIE_GBE_H	RO	0x0000XXXX	32	See SYS_PCIE_GBE Register .
0x0084	SYS_PROC_ID0	RW	0x0X000000	32	See SYS_PROC_ID0 Register .
0x0120	SYS_FAN_SPEED	RW	0x00000000	32	See SYS_FAN_SPEED Register .

5.6.2 SYS_ID Register

The SYS_ID Register characteristics are:

Purpose

Contains information about the Morello SDP and the bus and image versions inside the IOFPGA.

Usage constraints

The SYS_ID Register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-136: SYS_ID Register bit assignments

Bits	Name	Type	Function
[31:28]	Rev	RO	Board revision: 0x0: Rev A board. This is the prototype board and contains the Morello SoC.
[26:16]	HBI	RO	HBI board number in BCD: 0x316: HBI0316.
[15:12]	Build	RO	Build variant of board: 0xF: All builds.

Bits	Name	Type	Function
[11:8]	Arch	RO	IOFPGA bus architecture: 0x4: AHB. 0x5: AXI.
[7:0]	FPGA	RO	FPGA build in BCD. The actual value that is read depends on the FPGA build.

5.6.3 SYS_SW Register

The SYS_SW Register characteristics are:

Purpose

Stores the 8 user DIP switches on the Morello board. A bit set to 1 indicates that the switch is ON.

Usage constraints

Bits[29:28] are read-only. Bits[7:0] are read/write.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-137: SYS_SW Register bit assignments

Bits	Name	Type	Function
[31:30]	-	-	Reserved.
[29]	nUART0CTS	RO	UART0 CTS signal.
[28]	nUART0DSR	RO	UART0 DSR signal.
[27:8]	-	-	Reserved.
[7:0]	HARDWARE_USER_SW.	RW	State of the 8 user DIP switches on the board. Application software can read these switch settings: 0b0: OFF. 0b1: ON.

5.6.4 SYS_LED Register

The SYS_LED Register characteristics are:

Purpose

Controls the eight user LEDs on the Morello board. All LEDs are turned OFF at reset. The Boot Monitor updates the LED value.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-138: SYS_LED Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	LED[7:0]	RW	Set or read the user LED states: 0b0: OFF. 0b1: ON.

5.6.5 SYS_100HZ Register

The SYS_100HZ Register characteristics are:

Purpose

A 32-bit counter that updates at 100Hz. The input clock derives from the 24MHz clock generator on the Morello board.

Usage constraints

The SYS_100HZ Register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-139: SYS_100HZ Register bit assignments

Bits	Name	Type	Function
[31:0]	100HZ_COUNT	RO	Contains the count, at 100Hz, since the last CB_nRST reset.

5.6.6 SYS_FLAG Registers

The SYS_FLAG Registers characteristics are:

Purpose

Provide two 32-bit registers, SYS_FLAGS and SYS_NVFLAGS, that contain general-purpose flags. The application software defines the meaning of the flags. You use the SYS_FLAGSSET, SYS_FLAGSCLR, SYS_NVFLAGSSET, and SYS_NVFLAGSCLR registers to set and clear the bits in the Flag Registers.

Usage constraints

The SYS_FLAGS and SYS_NVFLAGS Registers are read-only.

The SYS_FLAGSSET, SYS_FLAGSCLR, SYS_NVFLAGSSET, and SYS_NVFLAGSCLR Registers are write-only.

Configurations

Available in all Morello SDP configurations.

SYS_FLAGS Register

The SYS_FLAGS Register is one of the two flag registers. It contains the current states of the flags.

The SYS_FLAGS Register is volatile, that is, a reset signal from the reset push button resets the SYS_FLAGS Register.

You use the SYS_FLAGSSET Register to set bits in the SYS_FLAGS Register. Write 1 to set the associated flag. Write 0 to leave the associated flag unchanged.

You use the SYS_FLAGSCLR Register to clear bits in the SYS_FLAGS Register. Write 1 to clear the associated flag. Write 0 to leave the associated flag unchanged.

SYS_NVFLAGS Register

The SYS_NVFLAGS Register is one of the two flag registers. It contains the current states of the flags.

The SYS_NVFLAGS Register is non-volatile, that is, a reset signal from the reset push button does not reset the SYS_FLAGS Register. Only **CB_nPOR** resets the SYS_NVFLAGS Register.

You use the SYS_NVFLAGSSET Register to set bits in the SYS_NVFLAGS Register. Write 1 to set the associated flag. Write 0 to leave the associated flag unchanged.

You use the SYS_NVFLAGSCLR Register to clear bits in the SYS_NVFLAGS Register. Write 1 to clear the associated flag. Write 0 to leave the associated flag unchanged.

5.6.7 SYS_CFGSW Register

The SYS_CFGSW Register characteristics are:

Purpose

Contains the value of *CONFSWITCH* in the `config.txt` file.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-140: SYS_CFGSW Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved.
[7:0]	SOFT_CONFIG_SWITCH	RW	Software applications can read these switch settings. The application software defines the meanings of the switch settings. The reset signals set these bits to the value of <i>CONFSWITCH</i> in the <code>config.txt</code> file. Note: The configuration system does not use the contents of this register for board configuration.

5.6.8 SYS_24MHZ Register

The SYS_24MHZ Register characteristics are:

Purpose

A 32-bit counter that updates at 24MHz. The clock source is the 24MHz clock generator on the Morello SDP.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-141: SYS_24MHZ Register bit assignments

Bits	Name	Type	Function
[31:0]	24MHZ_COUNT	RO	Contains the count, at 24MHz, since the last CB_nRST reset. CB_nRST sets the register to zero and then the count resumes.

5.6.9 SYS_PCIE_CNTL Register

The SYS_PCIE_CNTL Register characteristics are:

Purpose

Error signal from PCIe switch and reset signal to PCIe Express slots.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-142: SYS_PCIE_CNTL Register bit assignments

Bits	Name	Type	Function
[31:2]	-	-	Reserved.
[1]	PCIE_RSTHALT	RW	Error signal from PCIe switch.
[0]	PCIE_nPERST	RW	Reset signal to PCIe expansion slots.

5.6.10 SYS_PCIE_GBE Register

The SYS_PCIE_GBE Register characteristics are:

Purpose

Contains the 48-bit PCI Express Ethernet MAC address.

Usage constraints

Bits[47:0] are read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-143: SYS_PCI_GBE Register bit assignments

Bits	Name	Type	Function
[63:48]	-	-	Reserved.
[47:32]	SYS_PCIE_GBE_H	RO	Most significant 16 bits of the PCI Express Ethernet MAC address.
[31:0]	SYS_PCIE_GBE_L	RO	Least significant 32 bits of the PCI Express Ethernet MAC address.

5.6.11 SYS_PROC_ID0 Register

The SYS_PROC_ID0 Register characteristics are:

Purpose

Identifies the active clusters in the Morello SoC.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-144: SYS_PROC_ID0 Register bit assignments

Bits	Name	Type	Function
[31:24]	PROC_ID0	RW	Denotes active clusters.
[23:0]	-	-	Reserved.

5.6.12 SYS_FAN_SPEED Register

The SYS_FAN_SPEED Register characteristics are:

Purpose

Contains a value that represents the fan operating speed. The MCC uses this value to moderate the speed of the cooling fan on the Morello SDP.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-145: SYS_FAN_SPEED Register bit assignments

Bits	Name	Type	Function
[31]	UPD_ATE_FAN_SPEED	RW	Set this bit to 1 when updating the fan speed control bits [4:0]. The system clears this bit to 0 after updating the fan speed. The default value is 0b0.
[30:5]	-	-	Reserved.
[4:0]	FAN_SPEED	RW	Indicates and controls the speed of the board cooling fan. The fan has 30 speed settings: 0b00010: Minimum fan speed. 0b11111: Maximum fan speed. Note: 0b00000 and 0b00001 are invalid settings. Do not use them.

5.6.13 SP810_CTRL Register

The SP810_CTRL Register characteristics are:

Purpose

This register in the SP810 system controller selects the source clocks for the four SP804 timers in the IOFPGA.

Usage constraints

There are no usage constraints.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB system register summary](#).

The following table shows the bit assignments.

Table 5-146: SP810_CTRL Register bit assignments

Bits	Name	Type	Function
[31:22]	-		Reserved.
[21]	TimerEn3Sel		Selects the source clock for SP804 3 timer clock TIM_CLK[3]: 0b0 : TIM_CLK[3] = 32kHz. 0b1 : TIM_CLK[3] = 1MHz. Note: The default is 0b0.
[20]	-		Reserved.

Bits	Name	Type	Function
[19]	TimerEn2Sel		<p>Selects the source clock for SP804 2 timer clock TIM_CLK[2]:</p> <p>0b0 : TIM_CLK[2] = 32kHz.</p> <p>0b1 : TIM_CLK[2] = 1MHz.</p> <p>Note: The default is 0b0.</p>
[18]	-		Reserved.
[17]	TimerEn1Sel		<p>Selects the source clock for SP804 1 timer clock TIM_CLK[1]:</p> <p>0b0 : TIM_CLK[1] = 32kHz.</p> <p>0b1 : TIM_CLK[1] = 1MHz.</p> <p>Note: The default is 0b0.</p>
[16]	-		Reserved.
[15]	TimerEn0Sel		<p>Selects the source clock for SP804 0 timer clock TIM_CLK[0]:</p> <p>0b0 : TIM_CLK[0] = 32kHz.</p> <p>0b1 : TIM_CLK[0] = 1MHz.</p> <p>Note: The default is 0b0.</p>
[14:0]	-		Reserved.

5.7 Display clock and power control logic registers

Morello contains the following display clock and power control logic registers.

5.7.1 Display clock and power control logic registers summary

The base memory address of the display clock and power control logic registers in Morello is 0x5062_0000, in the Element management peripherals region of the SCP memory map.

The following table shows the display clock and power control logic registers in offset order from the base memory address. Undefined registers are reserved. Software must not attempt to access these registers.

Base memory address

0x5062_0000

Table 5-147: Display clock and power control logic registers summary

Offset	Name	Type	Reset	Width	Description
0x000 - 0x82C	-	RAZ/ WI	-	-	Reserved.
0x830	DISPLAYCLK_CTRL	RW	0x0000_0001	32	See DISPLAYCLK_CTRL Register .
0x834	DISPLAYCLK_DIV1	RW/ RO	0x0000_001F	32	See DISPLAYCLK_DIV1 Register .
0x838	DISPLAYCLK_DIV2	RW/ RO	0x0000_001F	32	See DISPLAYCLK_DIV2 Register . Control register
0x83C - 0x9FC	-	RAZ/ WI	-	-	Reserved.
0xA00	CLKFORCE_STATUS	RO	-	32	See CLKFORCE_STATUS Register .
0xA04	CLKFORCE_SET	WO	-	32	See CLKFORCE_SET Register .
0xA08	CLKFORCE_CLR	WO	0x0	32	See CLKFORCE_CLR Register .
0xA0C - 0xFBC	-	RAZ/ WI	-	-	Reserved.
0xFC0	PIK_POWER_CONTROL_LOGIC	RO	0x0052_0001	32	See PIK_POWER_CONTROL_LOGIC Register .
0xFD0	PID4	RO	0x44	8	See PID4 Register .
0xFD4	-	RAZ/ WI	-	-	Reserved.
0xFD8	-	RAZ/ WI	-	-	Reserved.
0xFDC	-	RAZ/ WI	-	-	Reserved.
0xFE0	PID0	RO	0xB8	8	See PID0 Register .
0xFE4	PID1	RO	0xB0	8	See PID1 Register .
0xFE8	PID2	RO	0x0B	8	See PID2 Register .
0xFEC	PID3	RO	0x00	8	See PID3 Register .
0xFF0	ID0	RO	0x0D	8	See Component ID0 Register .
0xFF4	ID1	RO	0xF0	8	See Component ID1 Register .
0xFF8	ID2	RO	0x05	8	See Component ID2 Register .
0xFFC	ID3	RO	0xB1	8	See Component ID3 Register .
0x1000 - 0x1FFC	DPU-PPU0	-	-	-	PPU Configuration dependent registers. For the DPU-PPU0 register descriptions, see <i>Arm® Power Policy Unit Architecture Specification</i> .

5.7.2 DISPLAYCLK_CTRL Register

The DISPLAYCLK_CTRL Register characteristics are:

Purpose

Controls the display clock register.

Usage constraints

This register is read-write.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the DISPLAYCLK_CTRL Register bit assignments.

Table 5-148: DISPLAYCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:24]	ENTRY_DLY	RW	Number of clock cycles between the clock not being required and the request to dynamically clock gate it: 0x0 - No cycles 0x1 - 1 cycle ... 0xFF - 255 cycles
[23:16]	-	-	Reserved.
[15:8]	CLKSELECT_CUR	RO	Acknowledges the currently selected clock source: 0x00 - Clock Gated 0x01 - REFCLK 0x02 - DISPLAYPLLCLK 0x04 - SYSPLLCLK Other values are Reserved.
[7:0]	CLKSELECT	RW	Selects the clock source: 0x00 - Clock Gated 0x01 - REFCLK 0x02 - DISPLAYPLLCLK 0x04 - SYSPLLCLK Other values are Reserved. The result of writing one of the Reserved values into this field is UNPREDICTABLE .

5.7.3 DISPLAYCLK_DIV1 Register

The DISPLAYCLK_DIV1 Register characteristics are:

Purpose

Controls the display clock divider 1 register.

Usage constraints

This register is read-write and read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the DISPLAYCLK_DIV1 Register bit assignments.

Table 5-149: DISPLAYCLK_DIV1 Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Acknowledges the currently selected clock divider value for the clock source selected by bit n. For example, *CLK_DIV1 acknowledges the divider value for the clock selected when CLKSELECT is 0x02. Currently selected clock divider value for the clock source selected by register DISPLAYCLK_CTRL. Clock divider value = CLKDIV_CUR + 1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Clock divider value for the clock source selected by register DISPLAYCLK_CTR. Clock divider value = CLKDIV + 1.

5.7.4 DISPLAYCLK_DIV2 Register

The DISPLAYCLK_DIV2 Register characteristics are:

Purpose

Controls the display clock divider 2 register.

Usage constraints

This register is read-write and read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the DISPLAYCLK_DIV2 Register bit assignments.

Table 5-150: DISPLAYCLK_DIV2 Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Acknowledges the currently selected clock divider value for the clock source selected by bit n. For example, *CLK_DIV2 acknowledges the divider value for the clock selected when CLKSELECT is 0x02. Currently selected clock divider value for the clock source selected by register DISPLAYCLK_CTRL. Clock divider value = CLKDIV_CUR + 1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	Clock divider value for the clock source selected by register DISPLAYCLK_CTR. Clock divider value = CLKDIV + 1.

5.7.5 CLKFORCE_STATUS Register

The bit allocation is the same as the CLKFORCE_SET register. If a bit reads as 1 then the associated dynamic clock gating is disabled, otherwise it is enabled. The CLKFORCE_STATUS Register characteristics are:

Purpose

Clock force status register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the CLKFORCE_STATUS Register bit assignments.

Table 5-151: CLKFORCE_STATUS Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3]	ACLKDPFORCE_ STATUS	RO	Clock status for ACLKDP.
[2:0]	-	-	Reserved.

5.7.6 CLKFORCE_SET Register

Writing 1 to a bit within the CLKFORCE_SET register disables any dynamic hardware clock gating, while writing 0 to a bit is ignored. The CLKFORCE_SET Register characteristics are:

Purpose

Clock force set register.

Usage constraints

This register is write-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the CLKFORCE_SET Register bit assignments.

Table 5-152: CLKFORCE_SET Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3]	ACLKDPFORCE_SET	WO	Write 1'b1 to enable clock force.
[2:0]	-	-	Reserved.

5.7.7 CLKFORCE_CLR Register

The bit allocation is the same as the CLKFORCE_SET register. Writing 1 to a bit within the CLKFORCE_CLR register enables the dynamic hardware clocking gating, while writing 0 to a bit is ignored. The CLKFORCE_CLR Register characteristics are:

Purpose

Clock force clear register.

Usage constraints

This register is write-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the CLKFORCE_CLR Register bit assignments.

Table 5-153: CLKFORCE_CLR Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3]	ACLKDPFORCE_CLR	WO	Write 1'b1 to clear (disable) clock force.
[2:0]	-	-	Reserved.

5.7.8 PIK_POWER_CONTROL_LOGIC Register

The PIK_POWER_CONTROL_LOGIC configuration register characteristics are:

Purpose

Power control logic configuration register. The value is dependent upon chosen configuration.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the PIK_POWER_CONTROL_LOGIC Register bit assignments.

Table 5-154: PIK_POWER_CONTROL_LOGIC Register bit assignments

Bits	Name	Type	Function
[31:16]	-	RO	POWER CONTROL LOGIC_ID. It is set to 0x0052.
[15:4]	-	-	Reserved
[3:0]	no_of_ppu	RO	Defines the number of PPU's in the POWER CONTROL LOGIC. This value is set to 0x1 to indicate one PPU.

5.7.9 PID4 Register

The PID4 register characteristics are:

Purpose

Peripheral ID 4 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the PID4 Register bit assignments.

Table 5-155: PID4 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ.
[7:4]	4KB_count	RO	The number of 4KB address blocks required to access the registers, expressed in powers of 2. These bits read back as 0x4. This means that the POWER CONTROL LOGIC occupies 64KB address block.
[3:0]	jep106_c_code	RO	The JEP106 continuation code value represents how many 0x7F continuation characters occur in the identity code of the manufacturer. These bits read back as 0x4.

5.7.10 PID0 Register

The PID0 register characteristics are:

Purpose

Peripheral ID 0 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the PID0 Register bit assignments.

Table 5-156: PID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ.
[7:0]	part_number_0	RO	These bits read back as 0xB8.

5.7.11 PID1 Register

The PID1 register characteristics are:

Purpose

Peripheral ID 1 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the PID1 Register bit assignments.

Table 5-157: PID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:4]	jep106_id_3_0	RO	JEP106 identity code [3:0]. See the <i>JEDEC Standard Manufacturer's Identification Code</i> .
[3:0]	part_number_1	RO	These bits read back as 0x0.

5.7.12 PID2 Register

The PID2 register characteristics are:

Purpose

Peripheral ID 2 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the PID2 Register bit assignments.

Table 5-158: PID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:4]	Revision	RO	Identifies the revision of the base POWER CONTROL LOGIC. For revision r0p0, this field is set to 0x0.
[3]	jedec_used	RO	This indicates that the POWER CONTROL LOGIC uses a <i>manufacturer's identity code</i> that was allocated by JEDEC according to JEP106. This bit always reads back as 0x1.
[2:0]	jep106_id_6_4	RO	JEP106 identity code [6:4].

5.7.13 PID3 Register

The PID3 register characteristics are:

Purpose

Peripheral ID 3 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the PID3 Register bit assignments.

Table 5-159: PID3 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:4]	RevAnd	RO	The top-level RTL provides a 4-bit input, ECOREVNUM, that is normally tied LOW and provides a read value of 0x0. When silicon is available, and if metal fixes are necessary, the manufacturer can modify the tie-offs to indicate a revision of the silicon.
[3:0]	mod_number	RO	This is set to 0x0.

5.7.14 Component ID0 Register

The ID0 register characteristics are:

Purpose

Component ID 0 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the Component ID0 Register bit assignments.

Table 5-160: Component ID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_0	RO	These bits read back as 0x0D.

5.7.15 Component ID1 Register

The ID1 register characteristics are:

Purpose

Component ID 1 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the Component ID1 Register bit assignments.

Table 5-161: Component ID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_1	RO	These bits read back as 0xF0.

5.7.16 Component ID2 Register

The ID2 register characteristics are:

Purpose

Component ID 2 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the Component ID2 Register bit assignments.

Table 5-162: Component ID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_2	RO	These bits read back as 0x05.

5.7.17 Component ID3 Register

The ID3 register characteristics are:

Purpose

Component ID 3 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Display clock and power control logic registers summary](#).

The following table shows the Component ID3 Register bit assignments.

Table 5-163: Component ID3 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_2	RO	These bits read back as 0xB1.

5.8 GPU clock and power control logic registers

Morello contains the following GPU clock and power control logic registers.

5.8.1 GPU clock and power control logic registers summary

The base memory address of the GPU clock and power control logic registers in Morello is 0x5063_0000, in the Element management peripherals region of the SCP memory map.

The following table shows the GPU clock and power control logic registers in offset order from the base memory address. Undefined registers are reserved. Software must not attempt to access these registers.

Base memory address

0x5063_0000

Table 5-164: GPU clock and power control logic registers summary

Offset	Name	Type	Reset	Width	Description
0x000 - 0x7FC	-	RAZ/ WI	-	-	Reserved.
0x800 - 0x80C	-	RAZ/ WI	-	-	Reserved.
0x810	GPUCLK_CTRL	RW	0x0000_0001	32	See GPUCLK_CTRL Register .
0x814	GPUCLK_DIV1	RW	0x0000_001F	32	See GPUCLK_DIV1 Register .
0x818	GPUCLK_DIV2	RW	0x0000_001F	32	See GPUCLK_DIV2 Register .
0x81C - 0x820	-	RAZ/ WI	-	-	Reserved.
0x820	ACLKGPU_CTRL	RW	0x0000_00F0	32	See ACLKGPU_CTRL Register .
0x824 - 0x9FC	-	RAZ/ WI	-	-	Reserved.
0xA00	CLKFORCE_STATUS	RO	-	32	See CLKFORCE_STATUS Register .
0xA04	CLKFORCE_SET	WO	-	32	See CLKFORCE_SET Register .
0xA08	CLKFORCE_CLR	WO	0x0	32	See CLKFORCE_CLR Register .
0xA0C - 0xFB8	-	RAZ/ WI	0x0	-	Reserved.
0xFBC	CAP	RO	-	32	See CAP Register .
0xFC0	PIK_POWER_CONTROL_LOGIC	RO	0x0041_0001	32	See PIK_POWER_CONTROL_LOGIC Register .
0xFD0	PID4	RO	0x44	8	See PID4 Register .
0xFD4 - 0xFD8	-	RAZ/ WI	-	-	Reserved.
0xFD8	-	RAZ/ WI	-	-	Reserved.
0xFDC - 0xFE0	-	RAZ/ WI	-	-	Reserved.
0xFE0	PID0	RO	0xB8	8	See PID0 Register .
0xFE4	PID1	RO	0xB0	8	See PID1 Register .
0xFE8	PID2	RO	0x0B	8	See PID2 Register .
0xFEC	PID3	RO	0x00	8	See PID3 Register .
0xFF0	ID0	RO	0x0D	8	See Component ID0 Register .
0xFF4	ID1	RO	0xF0	8	See Component ID1 Register .
0xFF8	ID2	RO	0x05	8	See Component ID2 Register .
0xFFC	ID3	RO	0xB1	8	See Component ID3 Register .
0x1000 - 0x1FFC	GPU-PPU0	-	-	-	PPU Configuration dependent registers. For the DPU-PPU0 register descriptions, see Arm® Power Policy Unit Architecture Specification .

5.8.2 GPUCLK_CTRL Register

The GPUCLK_CTRL Register characteristics are:

Purpose

Controls the GPU Clock Control register.

Usage constraints

This register is read-write.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the GPUCLK_CTRL Register bit assignments.

Table 5-165: GPUCLK_CTRL Register bit assignments

Bits	Name	Type	Function
[31:24]	ENTRY_DLY	RW	Number of clock cycles between the clock not being required and the request to dynamically clock gate it: 0x0 - No cycles 0x1 - 1 cycle ... 0xFF - 255 cycles This field is only available if CAP register bit 0 is set to a 1. Otherwise it is Reserved.
[23:16]	-	-	Reserved.
[15:8]	CLKSELECT_CUR	RO	Acknowledges the currently selected clock source: 0x00 - Clock Gated 0x01 - REFCLK 0x02 - SYSPLLCLK 0x04 - GPUPLLCLK Other values are Reserved.

Bits	Name	Type	Function
[7:0]	CLKSELECT	RW	<p>Selects the clock source:</p> <p>0x00 - Clock Gated</p> <p>0x01 - REFCLK</p> <p>0x02 - SYSPLLCLK</p> <p>0x04 - GPUPLLCLK</p> <p>Other values are Reserved. The result of writing one of the Reserved values into this field is UNPREDICTABLE.</p>

5.8.3 GPUCLK_DIV1 Register

The GPUCLK_DIV1 Register characteristics are:

Purpose

Controls the GPU clock divider control 1 register.

Usage constraints

This register is read-write and read only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the GPUCLK_DIV1 Register bit assignments.

Table 5-166: GPUCLK_DIV1 Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	<p>Acknowledges the currently selected clock divider value for the clock source selected by bit n. For example, GPUCLK_DIV1 acknowledges the divider value for the clock selected when CLKSELECT is 0x02.</p> <p>Currently selected clock divider value for the clock source selected by register GPUCLK_CTRL.</p> <p>Clock divider value = CLKDIV_CUR + 1.</p>
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	<p>GPUCLK_DIVn register requests a new clock divider value for the clock source selected by bit n. For example, GPUCLK_DIV1 selects the divider value for the clock selected when CLKSELECT is 0x02.</p> <p>Clock divider value = CLKDIV + 1.</p>

5.8.4 GPUCLK_DIV2 Register

The GPUCLK_DIV2 Register characteristics are:

Purpose

Controls the GPU clock divider control 2 register.

Usage constraints

This register is read-write and read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the GPUCLK_DIV2 Register bit assignments.

Table 5-167: GPUCLK_DIV2 Register bit assignments

Bits	Name	Type	Function
[31:21]	-	-	Reserved.
[20:16]	CLKDIV_CUR	RO	Acknowledges the currently selected clock divider value for the clock source selected by bit n. For example, GPUCLK_DIV2 acknowledges the divider value for the clock selected when CLKSELECT is 0x02. Currently selected clock divider value for the clock source selected by register GPUCLK_CTRL. Clock divider value = CLKDIV_CUR + 1.
[15:5]	-	-	Reserved.
[4:0]	CLKDIV	RW	GPUCLK_DIVn register requests a new clock divider value for the clock source selected by bit n. For example, GPUCLK_DIV2 selects the divider value for the clock selected when CLKSELECT is 0x02. Clock divider value = CLKDIV + 1.

5.8.5 ACLKGPU_CTRL Register

The ACLKGPU_CTRL Register characteristics are:

Purpose

Selects the divider ratio for ACLKGPU. The input clock is GPUCLK.

Usage constraints

This register is read-write and read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the ACLKGPU_CTRL Register bit assignments.

Table 5-168: ACLKGPU_CTRL Register bit assignments

Bits	Name	Type	Function
[31:16]	-	-	Reserved.
[15:12]	CLKDIV_CUR	RO	Acknowledges the currently active clock divider value. Clock divider value = CLKDIV_CUR + 1.
[11:8]	-	-	Reserved.
[7:4]	CLKDIV	RW	Requests new clock divider value. Clock divider value = CLKDIV + 1.
[3:0]	-	-	Reserved.

5.8.6 CLKFORCE_STATUS Register

The bit allocation is the same as the CLKFORCE_SET register. If a bit reads as 1 then the associated dynamic clock gating is disabled, otherwise it is enabled. The CLKFORCE_STATUS Register characteristics are:

Purpose

Clock force status register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the CLKFORCE_STATUS Register bit assignments.

Table 5-169: CLKFORCE_STATUS Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3]	ELACLKFORCE	RO	Clock status for ELA. This field is only available if CAP register bit 1 is set to a 1. Otherwise it is Reserved.
[2]	ACCLKGPUFORCE	RO	Clock status for ACLKGPU.

Bits	Name	Type	Function
[1]	GPUCLKFORCE	RO	Clock status for GPUCLK. This field is only available if CAP register bit 0 is set to a 1. Otherwise it is Reserved.
[0]	-	-	Reserved.

5.8.7 CLKFORCE_SET Register

Writing 1 to a bit within the CLKFORCE_SET register disables any dynamic hardware clock gating, while writing 0 to a bit is ignored. The CLKFORCE_SET Register characteristics are:

Purpose

Clock force set register.

Usage constraints

This register is write-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the CLKFORCE_SET Register bit assignments.

Table 5-170: CLKFORCE_SET Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3]	ELACLKFORCE	WO	Write 0b1 to enable clock force for ELA. This field is only available if CAP register bit 1 is set to a 1. Otherwise it is Reserved.
[2]	ACLKGPUFORCE	WO	Write 0b1 to enable clock force for ACLKGPU.
[1]	GPUCLKFORCE	WO	Write 0b1 to enable clock force for GPUCLK. This field is only available if CAP register bit 0 is set to a 1. Otherwise it is Reserved.
[0]	-	-	Reserved.

5.8.8 CLKFORCE_CLR Register

The bit allocation is the same as the CLKFORCE_SET register. Writing 1 to a bit within the CLKFORCE_CLR register enables the dynamic hardware clocking gating, while writing 0 to a bit is ignored. The CLKFORCE_CLR Register characteristics are:

Purpose

Clock force clear register.

Usage constraints

This register is write-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the CLKFORCE_CLR Register bit assignments.

Table 5-171: CLKFORCE_CLR Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3]	ELACKFORCE	WO	Write 0b1 to clear (disable) clock force for ELA. This field is only available if CAP register bit 1 is set to a 1. Otherwise it is Reserved.
[2]	ACKGPUFORCE	WO	Write 0b1 to clear (disable) clock force for ACKGPU.
[1]	GPUCLKFORCE	WO	Write 0b1 to clear (disable) clock force for GPUCLK. This field is only available if CAP register bit 0 is set to a 1. Otherwise it is Reserved.
[0]	-	-	Reserved.

5.8.9 CAP Register

The CAP Register characteristics are:

Purpose

Set CAP value based on the configuration.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the CAP Register bit assignments.

Table 5-172: CAP Register bit assignments

Bits	Name	Type	Function
[31:2]	-	-	Reserved.

Bits	Name	Type	Function
[1]	ELA support	RO	0b0 - ELA is not supported 0b1 - ELA is supported All the fields in the GPU Power Control registers pertaining to ELA are RAZ/WI if this bit is set to 0.
[0]	GPUCLK Gating	RO	0b0 - GPUCLK gating is not supported 0b1 - GPUCLK gating is supported.

5.8.10 PIK_POWER_CONTROL_LOGIC Register

The PIK_POWER_CONTROL_LOGIC configuration register characteristics are:

Purpose

Power control logic configuration register. The value is dependent upon chosen configuration.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the PIK_POWER_CONTROL_LOGIC Register bit assignments.

Table 5-173: PIK_POWER_CONTROL_LOGIC Register bit assignments

Bits	Name	Type	Function
[31:16]	-	RO	POWER CONTROL LOGIC_ID. It is set to 0x41.
[15:4]	-	-	Reserved
[3:0]	no_of_ppu	RO	Defines the number of PPU's in the POWER CONTROL LOGIC. This value is set to 0x1 to indicate one PPU.

5.8.11 PID4 Register

The PID4 register characteristics are:

Purpose

Peripheral ID 4 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the PID4 Register bit assignments.

Table 5-174: PID4 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ.
[7:4]	4KB_count	RO	The number of 4KB address blocks required to access the registers, expressed in powers of 2. These bits read back as 0x4. This means that the POWER CONTROL LOGIC occupies 64KB address block.
[3:0]	jep106_c_code	RO	The JEP106 continuation code value represents how many 0x7F continuation characters occur in the identity code of the manufacturer. These bits read back as 0x4.

5.8.12 PID0 Register

The PID0 register characteristics are:

Purpose

Peripheral ID 0 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the PID0 Register bit assignments.

Table 5-175: PID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ.
[7:0]	part_number_0	RO	These bits read back as 0xB8.

5.8.13 PID1 Register

The PID1 register characteristics are:

Purpose

Peripheral ID 1 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the PID1 Register bit assignments.

Table 5-176: PID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:4]	jep106_id_3_0	RO	JEP106 identity code [3:0]. See the <i>JEDEC Standard Manufacturer's Identification Code</i> .
[3:0]	part_number_1	RO	These bits read back as 0x0.

5.8.14 PID2 Register

The PID2 register characteristics are:

Purpose

Peripheral ID 2 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the PID2 Register bit assignments.

Table 5-177: PID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:4]	Revision	RO	Identifies the revision of the base POWER CONTROL LOGIC. For revision r0p0, this field is set to 0x0.
[3]	jedec_used	RO	This indicates that the POWER CONTROL LOGIC uses a <i>manufacturer's identity code</i> that was allocated by JEDEC according to JEP106. This bit always reads back as 0x1.
[2:0]	jep106_id_6_4	RO	JEP106 identity code [6:4]. See the <i>JEDEC Standard Manufacturer's Identification Code</i> .

5.8.15 PID3 Register

The PID3 register characteristics are:

Purpose

Peripheral ID 3 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the PID3 Register bit assignments.

Table 5-178: PID3 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:4]	RevAnd	RO	The top-level RTL provides a 4-bit input, ECOREVNUM, that is normally tied LOW and provides a read value of 0x0. When silicon is available, and if metal fixes are necessary, the manufacturer can modify the tie-offs to indicate a revision of the silicon.
[3:0]	mod_number	RO	This is set to 0x0.

5.8.16 Component ID0 Register

The ID0 register characteristics are:

Purpose

Component ID 0 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the Component ID0 Register bit assignments.

Table 5-179: Component ID0 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_0	RO	These bits read back as 0x0D.

5.8.17 Component ID1 Register

The ID1 register characteristics are:

Purpose

Component ID 1 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the Component ID1 Register bit assignments.

Table 5-180: Component ID1 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_1	RO	These bits read back as 0xF0.

5.8.18 Component ID2 Register

The ID2 register characteristics are:

Purpose

Component ID 2 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the Component ID2 Register bit assignments.

Table 5-181: Component ID2 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_2	RO	These bits read back as 0x05.

5.8.19 Component ID3 Register

The ID3 register characteristics are:

Purpose

Component ID 3 register.

Usage constraints

This register is read-only.

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [GPU clock and power control logic registers summary](#).

The following table shows the Component ID3 Register bit assignments.

Table 5-182: Component ID3 Register bit assignments

Bits	Name	Type	Function
[31:8]	-	-	Reserved, SBZ
[7:0]	comp_id_2	RO	These bits read back as 0xB1.

5.9 Generic Timer registers

Morello supports timer frames and synchronization for multichip mode.

5.9.1 Generic Timer registers summary

The following table shows the Generic Timer registers in offset order from the base memory address. Undefined registers are reserved. Software must not attempt to access these registers.

Table 5-183: Generic Timer registers summary

Offset	Name	Type	Reset	Width	Description
-	MST_GCNT_SYNC_CTRL	-	-	32	See MST_GCNT_SYNC_CTRL Register .
-	LVCHIP_GCNT_SYNC_CTRL	-	-	32	See SLVCHIP_GCNT_SYNC_CTRL Register .
-	SLVCHIP_GCNT_INT_STATUS	-	-	32	See SLVCHIP_GCNT_INT_STATUS Register .
0x0028	SLVCHIP_GCNT_RETRY_CNT	RW	0x0	32	See SLVCHIP_GCNT_RETRY_CNT Register .

5.9.2 MST_GCNT_SYNC_CTRL Register

The MST_GCNT_SYNC_CTRL Register characteristics are:

Purpose

-

Usage constraints

-

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Generic Timer registers summary](#).

The following table shows the MST_GCNT_SYNC_CTRL Register bit assignments.

Table 5-184: MST_GCNT_SYNC_CTRL Register bit assignments

Bits	Name	Type	Function
[31:7]	-	-	Reserved.
[6:2]	TURN_AR_TIME	-	Number of clocks to wait between back to back start of the transactions as noted in the diagram attached
[1]	EN_SYNC_IMM	-	Enables immediate synchronization process
[0]	EN	-	Enable Sync process

5.9.3 SLVCHIP_GCNT_SYNC_CTRL Register

The SLVCHIP_GCNT_SYNC_CT Register characteristics are:

Purpose

-

Usage constraints

-

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Generic Timer registers summary](#).

The following table shows the SLVCHIP_GCNT_SYNC_CT Register bit assignments.

Table 5-185: SLVCHIP_GCNT_SYNC_CT Register bit assignments

Bits	Name	Type	Function
[31:7]	-	-	Reserved.
[6:2]	TURN_AR_TIME	-	Number of clocks to wait between back to back start of the transactions for retries.
[1]	EN_SYNC_IMM	-	Enable immediate synchronization process.
[0]	EN	-	Enable synchronization process.

5.9.4 SLVCHIP_GCNT_INT_STATUS Register

The SLVCHIP_GCNT_INT_STATUS Register characteristics are:

Purpose

-

Usage constraints

-

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Generic Timer registers summary](#).

The following table shows the SLVCHIP_GCNT_INT_STATUS Register bit assignments.

Table 5-186: SLVCHIP_GCNT_INT_STATUS Register bit assignments

Bits	Name	Type	Function
[31:4]	-	-	Reserved.
[3]	Time out Interrupt	-	Time out interrupt when manager does not respond to sync request.
[2]	Sync failed interrupt status	-	Synchronization failed when software request by setting EN_SYNC_IMM register field.
[1]	Sync passed interrupt status	-	Synchronization passed when software request by setting EN_SYNC_IMM register field
[0]	Sync failed after number of retries specified	-	Synchronization failed after n-number of retries.

5.9.5 SLVCHIP_GCNT_RETRY_CNT Register

The SLVCHIP_GCNT_RETRY_CNT Register characteristics are:

Purpose

-

Usage constraints

-

Configurations

Available in all Morello configurations.

Memory offset and full register reset value

See [Generic Timer registers summary](#).

The following table shows the SLVCHIP_GCNT_RETRY_CNT Register bit assignments.

Table 5-187: SLVCHIP_GCNT_RETRY_CNT Register bit assignments

Bits	Name	Type	Function
[31:5]	-	-	Reserved.
[4:0]	RETRY_COUNT	-	Retries before raising the interrupt and go into stall mode.

5.10 APB energy meter registers

The IOFPGA contains the APB energy meter registers.

5.10.1 APB energy meter registers summary

The base memory address of the APB energy meter registers in the IOFPGA is 0x1C01_0000.

The APB energy meter registers contain values that represent supply currents, supply voltages, and power consumption in the Morello SoC.

The IOFPGA energy registers relate to the following parts of the Morello SoC:

- Morello clusters, 1 and 0.
- CCIX and PCIe PHY cluster.
- The fabric of the Morello SoC outside the clusters, that is, the parts of the chip that operate from the VSYS power supply.

The IOFPGA energy registers measure the following values of each block:

- Instantaneous current consumption.
- Instantaneous voltage supplies.
- Instantaneous power consumption.
- Cumulative energy consumption.
- Morello clusters, 1 and 0.
- CCIX and PCIe PHY cluster.
- The fabric of the Morello SoC outside the clusters, that is, the parts of the chip that operate from the VSYS power supply.



The current, power, and energy meter registers are provisional and subject to characterization on the RevB boards.

The following table shows the registers in address offset order from the base memory address.

Offset	Name	Type	Reset	Width	Description
00D0	SYS_I_SYS	RO	0x0000_0000	32	See SYS_I_SYS Register .
00D4	SYS_I_CL0	RO	0x0000_0000	32	See SYS_I_CL0 Register .
00D8	SYS_I_PCIE	RO	0x0000_0000	32	See SYS_I_PCIE Register .
00DC	SYS_I_CL1	RO	0x0000_0000	32	See SYS_I_CL1 Register .
00E0	SYS_V_SYS	RO	0x0000_0000	32	See SYS_V_SYS Register .
00E4	SYS_V_CL0	RO	0x0000_0000	32	See SYS_V_CL0 Register .
00E8	SYS_V_PCIE	RO	0x0000_0000	32	See SYS_V_PCIE Register .
00EC	SYS_V_CL1	RO	0x0000_0000	32	See SYS_V_CL1 Register .
00F0	SYS_POW_SYS	RO	0x0000_0000	32	See SYS_POW_SYS Register .
00F4	SYS_POW_CL0	RO	0x0000_0000	32	See SYS_POW_CL0 Register .
00F8	SYS_POW_PCIE	RO	0x0000_0000	32	See SYS_POW_PCIE Register .

Offset	Name	Type	Reset	Width	Description
00FC	SYS_POW_CL1	RO	0x0000_0000	32	See SYS_POW_CL1 Register .
0100	SYS_ENM_L_SYS	RW	0x0000_0000	32	See SYS_ENM_SYS Register .
0104	SYS_ENM_H_SYS	RW	0x0000_0000	32	See SYS_ENM_SYS Register .
0108	SYS_ENM_L_CL0	RW	0x0000_0000	32	See SYS_ENM_CL0 Register .
010C	SYS_ENM_H_CL0	RW	0x0000_0000	32	See SYS_ENM_CL0 Register .
0110	SYS_ENM_L_PCIE		0x0000_0000	32	See SYS_ENM_PCIE Register .
0114	SYS_ENM_H_PCIE	RW	0x0000_0000	32	See SYS_ENM_PCIE Register .
0118	SYS_ENM_L_CL1	RW	0x0000_0000	32	See SYS_ENM_CL1 Register .
011C	SYS_ENM_H_CL1	RW	0x0000_0000	32	See SYS_ENM_CL1 Register .
0120	SYS_I_DDR0	RO	0x0000_0000	32	See SYS_I_DDR0 Register .
0124	SYS_I_DDR1	RO	0x0000_0000	32	See SYS_I_DDR1 Register .
0128	SYS_V_DDR0	RO	0x0000_0000	32	See SYS_V_DDR0 Register .
012C	SYS_V_DDR1	RO	0x0000_0000	32	See SYS_V_DDR1 Register .
0130	SYS_POW_DDR0	RO	0x0000_0000	32	See SYS_POW_DDR0 Register .
0134	SYS_POW_DDR1	RO	0x0000_0000	32	See SYS_POW_DDR1 Register .
0138	SYS_ENM_L_DDR0	RW	0x0000_0000	32	See SYS_ENM_DDR0 Register .
013C	SYS_ENM_H_DDR0	RW	0x0000_0000	32	See SYS_ENM_DDR0 Register .
0140	SYS_ENM_L_DDR1	RW	0x0000_0000	32	See SYS_ENM_DDR1 Register .
0144	SYS_ENM_H_DDR1	RW	0x0000_0000	32	See SYS_ENM_DDR1 Register .

5.10.2 SYS_I_SYS Register

The SYS_I_SYS Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous current consumption of the parts of the Morello SoC, outside the clusters, that operate from the VSYS power supply.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



Note

The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-189: SYS_I_SYS Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_I_SYS	RO	<p>12-bit representation of the instantaneous current consumption of the parts of the Morello SoC, outside the clusters, that operate from the VSYS power supply:</p> <ul style="list-style-type: none"> Full scale measurement, 4096, represents 5A. Full scale is 0xFFF. Measured current = (SYS_I_SYS+1)/761 amperes. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.3 SYS_I_CL0 Register

SYS_I_CL0 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous current consumption of Morello cluster 0.

Usage constraints

This register is read-only. You must use one of the cluster 1 cores to read this register.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-190: SYS_I_CL0 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_I_CL0	RO	<p>12-bit representation of the current consumption of Morello cluster 0:</p> <ul style="list-style-type: none"> Full scale measurement, 4096, represents 10A. Full scale is 0xFFF. Measured current = (SYS_I_CL0+1)/381 amperes. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.4 SYS_I_PCIE Register

The SYS_I_PCIE Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous current consumption of the PCIe cluster.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-191: SYS_I_PCIE Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_I_PCIE	RO	12-bit representation of the instantaneous current consumption of the PCIe cluster: <ul style="list-style-type: none"> Full scale measurement, 4096, represents 10A. Full scale is 0xFFF. Measured current = (SYS_I_PCIE+1)/381 amperes. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.5 SYS_I_CL1 Register

The SYS_I_CL1 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous current consumption of Morello cluster 1.

Usage constraints

This register is read-only. You must use one of the cluster 0 cores to read this register.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-192: SYS_I_CL1 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_I_CL1	RO	12-bit representation of the current consumption of Morello cluster 1: <ul style="list-style-type: none"> Full scale measurement, 4096, represents 5A. Full scale is 0xFFF. Measured current = (SYS_I_CL1+1)/761 amperes. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.6 SYS_V_SYS Register

The SYS_V_SYS Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous supply voltage of the parts of the Morello SoC, outside the clusters, that operate from the VSYS power supply.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).

The following table shows the bit assignments.

Table 5-193: SYS_V_SYS Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.

Bits	Name	Type	Function
[11:0]	SYS_V_SYS	RO	<p>12-bit representation of the instantaneous supply voltage of the parts of the Morello SoC, outside the clusters, that operate from the VSYS power supply:</p> <ul style="list-style-type: none"> Full scale measurement, 4096, represents 2V5. Full scale is 0xFFF. Measured voltage = (SYS_V_SYS+1)/1622 volts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.7 SYS_V_CL0 Register

The SYS_V_CL0 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous supply voltage of Morello SoC cluster 0.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).

The following table shows the bit assignments.

Table 5-194: SYS_V_CL0 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_V_CL0	RO	<p>12-bit representation of the instantaneous supply voltage of Morello SoC cluster 0:</p> <ul style="list-style-type: none"> Full scale measurement, 4096, represents 2V5. Full scale is 0xFFF. Measured voltage = (SYS_V_CL0+1)/1622 volts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.8 SYS_V_PCIE Register

The SYS_V_PCIE Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous supply voltage of the PCIe cluster.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).

The following table shows the bit assignments.

Table 5-195: SYS_V_PCIE Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_V_PCIE	RO	12-bit representation of the instantaneous supply voltage of the PCIe cluster: <ul style="list-style-type: none"> Full scale measurement, 4096, represents 2V5. Full scale is 0xFFF. Measured voltage = (SYS_V_PCIE+1)/1622 volts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.9 SYS_V_CL1 Register

The SYS_V_CL1 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous supply voltage of Morello SoC cluster 1.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).

The following table shows the bit assignments.

Table 5-196: SYS_V_CL1 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_V_CL1	RO	12-bit representation of the instantaneous supply voltage of Morello SoC cluster 1: <ul style="list-style-type: none"> Full scale measurement, 4096, represents 2V5. Full scale is 0xFFF. Measured voltage = (SYS_V_CL1+1)/1622 volts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.10 SYS_POW_SYS Register

The SYS_POW_SYS Register characteristics are:

Purpose

Contains a 24-bit representation of the instantaneous power consumption of the parts of the Morello SoC, outside the clusters, that operate from the VSYS power supply.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-197: SYS_POW_SYS Register bit assignments

Bits	Name	Type	Function
[31:24]	-	-	Reserved.
[23:0]	SYS_POW_SYS	RO	24-bit representation of the instantaneous power consumption of the parts of the SoC, outside the clusters, that operate from the VSYS power supply: <ul style="list-style-type: none"> The value of these bits represents $[\text{SYS_I_SYS(I)} \times \text{SYS_V_SYS(V)}]/1234803$ watts. Measured power consumption = $(\text{SYS_POW_SYS})/1234803$ The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.11 SYS_POW_CLO Register

The SYS_POW_CLO Register characteristics are:

Purpose

Contains a 24-bit representation of the instantaneous power consumption of Morello SoC cluster 0.

Usage constraints

This register is read-only. You must use one of the cluster 1 cores to read this register.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-198: SYS_POW_CL0 Register bit assignments

Bits	Name	Type	Function
[31:24]	-	-	Reserved.
[23:0]	SYS_POW_CL0	RO	24-bit representation of the instantaneous power consumption of Morello SoC cluster 0: <ul style="list-style-type: none"> The value of these bits represents $[\text{SYS_I_CL0(I)} \times \text{SYS_V_CL0(V)}]/617402$ watts. Measured power consumption = $(\text{SYS_POW_CL0})/617402$ watts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.12 SYS_POW_PCIE Register

The SYS_POW_PCIE Register characteristics are:

Purpose

Contains a 24-bit representation of the instantaneous power consumption of the PCIe cluster.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-199: SYS_POW_PCIE Register bit assignments

Bits	Name	Type	Function
[31:24]	-	-	Reserved.

Bits	Name	Type	Function
[23:0]	SYS_POW_PCIE	RO	24-bit representation of the instantaneous power consumption of the PCIe cluster: <ul style="list-style-type: none"> The value of these bits represents $[\text{SYS_I_PCIE(I)} \times \text{SYS_V_PCIE(V)}]/617402$ watts. Measured power consumption=$(\text{SYS_POW_PCIE})/617402$ watts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.13 SYS_POW_CL1 Register

The SYS_POW_CL1 Register characteristics are:

Purpose

Contains a 24-bit representation of the instantaneous power consumption of Morello SoC cluster 1.

Usage constraints

This register is read-only. You must use one of the cluster 0 cores to read this register.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-200: SYS_POW_CL1 Register bit assignments

Bits	Name	Type	Function
[31:24]	-	-	Reserved.
[23:0]	SYS_POW_CL1	RO	24-bit representation of the instantaneous power consumption of Morello SoC cluster 1: <ul style="list-style-type: none"> The value of these bits represents $[\text{SYS_I_CL1(I)} \times \text{SYS_V_CL1(V)}]/1234803$ watts. Measured power consumption=$(\text{SYS_POW_CL1})/1234803$ watts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.14 SYS_ENM_SYS Register

The SYS_ENM_SYS Register characteristics are:

Purpose

Contains a 64-bit representation of the accumulated energy consumption of the fabric of the Morello SoC outside the clusters.

Usage constraints

Writing to this register clears the 64-bit value.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-201: SYS_ENM_SYS Register bit assignments.

Bits	Name	Type	Function
[63:32]	SYS_ENM_H_SYS	RW	<p>Most significant 32 bits of a 64-bit representation of the accumulated energy consumption of the fabric of the Morello SoC outside the clusters:</p> <ul style="list-style-type: none"> The memory address offset of these bits is 0x0104. Accumulated energy = (SYS_ENM_CH0_H_SYS:SYS_ENM_L_SYS)/12348030000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.
[31:0]	SYS_ENM_L_SYS	RW	<p>Least significant 32 bits of a 64-bit representation of the accumulated energy consumption of the fabric of the Morello SoC outside the clusters:</p> <ul style="list-style-type: none"> The memory address offset of these bits is 0x0100. Accumulated energy = (SYS_ENM_CH0_H_SYS:SYS_ENM_L_SYS)/12348030000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.15 SYS_ENM_CLO Register

The SYS_ENM_CLO Register characteristics are:

Purpose

Contains a 64-bit representation of the accumulated energy consumption of the Morello SoC cluster 0.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-202: SYS_ENM_CLO Register bit assignments.

Bits	Name	Type	Function
[63:32]	SYS_ENM_H_CLO	RO	Most significant 32 bits of a 64-bit representation of the accumulated energy consumption of the Morello SoC cluster 0: <ul style="list-style-type: none"> The memory address offset of these bits is 0x010C. Accumulated energy = (SYS_ENM_H_CLO:SYS_ENM_L_CLO)/6174020000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.
[31:0]	SYS_ENM_L_CLO	RO	Least significant 32 bits of a 64-bit representation of the accumulated energy consumption of the Morello SoC cluster 0: <ul style="list-style-type: none"> The memory address offset of these bits is 0x0108. Accumulated energy = (SYS_ENM_H_CLO:SYS_ENM_L_CLO)/6174020000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.16 SYS_ENM_PCIE Register

The SYS_ENM_PCIE Register characteristics are:

Purpose

Contains a 64-bit representation of the accumulated energy consumption of the PCIe cluster 0.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-203: SYS_ENM_PCIE Register bit assignments.

Bits	Name	Type	Function
[63:32]	SYS_ENM_H_PCIE	RO	<p>Most significant 32 bits of a 64-bit representation of the accumulated energy consumption of the Morello SoC cluster 0:</p> <ul style="list-style-type: none"> The memory address offset of these bits is 0x0114. Accumulated energy = (SYS_ENM_H_PCIE:SYS_ENM_L_PCIE)/6174020000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.
[31:0]	SYS_ENM_L_PCIE	RO	<p>Least significant 32 bits of a 64-bit representation of the accumulated energy consumption of the Morello SoC cluster 0:</p> <ul style="list-style-type: none"> The memory address offset of these bits is 0x0110. Accumulated energy = (SYS_ENM_H_PCIE:SYS_ENM_L_PCIE)/6174020000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.17 SYS_ENM_CL1 Register

The SYS_ENM_CL1 Register characteristics are:

Purpose

Contains a 64-bit representation of the accumulated energy consumption of Morello SoC cluster 1.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-204: SYS_ENM_CL1 Register bit assignments.

Bits	Name	Type	Function
[63:32]	SYS_ENM_H_CL1	RO	<p>Most significant 32 bits of a 64-bit representation of the accumulated energy consumption of the Morello SoC cluster 1:</p> <ul style="list-style-type: none"> The memory address offset of these bits is 0x011C. Accumulated energy = (SYS_ENM_H_CL1:SYS_ENM_L_CL1)/12348030000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.
[31:0]	SYS_ENM_L_CL1	RO	<p>Least significant 32 bits of a 64-bit representation of the accumulated energy consumption of the Morello SoC cluster 1:</p> <ul style="list-style-type: none"> The memory address offset of these bits is 0x0118. Accumulated energy = (SYS_ENM_H_CL1:SYS_ENM_L_CL1)/12348030000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.18 SYS_I_DDR0 Register

SYS_I_DDR0 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous current consumption of DDR 0.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-205: SYS_I_DDR0 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_I_DDR0	RO	<p>12-bit representation of the instantaneous current consumption of DDR 0:</p> <ul style="list-style-type: none"> Full scale measurement, 4096, represents 10A. Full scale is 0xFFF. Measured current = (SYS_I_DDR0+1)/381 amperes. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.19 SYS_I_DDR1 Register

SYS_I_DDR1 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous current consumption of DDR 1.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-206: SYS_I_DDR1 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_I_DDR1	RO	12-bit representation of the instantaneous current consumption of DDR 1: <ul style="list-style-type: none"> Full scale measurement, 4096, represents 10A. Full scale is 0xFFF. Measured current = (SYS_I_DDR1+1)/381 amperes. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.20 SYS_V_DDR0 Register

The SYS_V_DDR0 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous supply voltage of DDR 0.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).

The following table shows the bit assignments.

Table 5-207: SYS_V_DDR0 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_V_DDR0	RO	12-bit representation of the instantaneous supply voltage of DDR 1: <ul style="list-style-type: none"> Full scale measurement, 4096, represents 2V5. Full scale is 0xFFF. Measured voltage = (SYS_V_DDR0+1)/1622 volts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.21 SYS_V_DDR1 Register

The SYS_V_DDR1 Register characteristics are:

Purpose

Contains a 12-bit representation of the instantaneous supply voltage of DDR 1.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).

The following table shows the bit assignments.

Table 5-208: SYS_V_DDR1 Register bit assignments

Bits	Name	Type	Function
[31:12]	-	-	Reserved.
[11:0]	SYS_V_DDR1	RO	12-bit representation of the instantaneous supply voltage of DDR 1: <ul style="list-style-type: none"> Full scale measurement, 4096, represents 2V5. Full scale is 0xFFF. Measured voltage = (SYS_V_DDR1+1)/1622 volts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.22 SYS_POW_DDR0 Register

The SYS_POW_DDR0 Register characteristics are:

Purpose

Contains a 24-bit representation of the instantaneous power consumption of DDR 0.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-209: SYS_POW_DDR0 Register bit assignments

Bits	Name	Type	Function
[31:24]	-	-	Reserved.
[23:0]	SYS_POW_DDR0	RO	24-bit representation of the instantaneous power consumption of DDR 0: <ul style="list-style-type: none"> The value of these bits represents $[\text{SYS_I_DDR0(I)} \times \text{SYS_V_DDR0(V)}]/617402$ watts. Measured power consumption = $(\text{SYS_POW_DDR0})/617402$ watts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.23 SYS_POW_DDR1 Register

The SYS_POW_DDR1 Register characteristics are:

Purpose

Contains a 24-bit representation of the instantaneous power consumption of DDR 1.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-210: SYS_POW_DDR1 Register bit assignments

Bits	Name	Type	Function
[31:24]	-	-	Reserved.
[23:0]	SYS_POW_DDR1	RO	24-bit representation of the instantaneous power consumption of DDR 1: <ul style="list-style-type: none"> The value of these bits represents $[\text{SYS_I_DDR1(I)} \times \text{SYS_V_DDR1(V)}]/617402$ watts. Measured power consumption = $(\text{SYS_POW_DDR1})/617402$ watts. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.24 SYS_ENM_DDR0 Register

The SYS_ENM_DDR0 Register characteristics are:

Purpose

Contains a 64-bit representation of the accumulated energy consumption of DDR 0.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-211: SYS_ENM_DDR0 Register bit assignments.

Bits	Name	Type	Function
[63:32]	SYS_ENM_H_DDR0	RO	Most significant 32 bits of a 64-bit representation of the accumulated energy consumption of DDR 0: <ul style="list-style-type: none"> The memory address offset of these bits is 0x013C. Accumulated energy = $(\text{SYS_ENM_H_DDR0}:\text{SYS_ENM_L_DDR0})/6174020000$ joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.
[31:0]	SYS_ENM_L_DDR0	RO	Least significant 32 bits of a 64-bit representation of the accumulated energy consumption of DDR 0: <ul style="list-style-type: none"> The memory address offset of these bits is 0x0138. Accumulated energy = $(\text{SYS_ENM_H_DDR0}:\text{SYS_ENM_L_DDR0})/6174020000$ joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.10.25 SYS_ENM_DDR1 Register

The SYS_ENM_DDR1 Register characteristics are:

Purpose

Contains a 64-bit representation of the accumulated energy consumption of DDR 1.

Usage constraints

This register is read-only.

Configurations

Available in all Morello board configurations.

Memory offset and full register reset value

See [APB energy meter registers summary](#).



The value measured by this register is provisional and subject to characterization on the RevB boards.

The following table shows the bit assignments.

Table 5-212: SYS_ENM_DDR1 Register bit assignments.

Bits	Name	Type	Function
[63:32]	SYS_ENM_H_DDR1	RO	Most significant 32 bits of a 64-bit representation of the accumulated energy consumption of DDR 1: <ul style="list-style-type: none"> The memory address offset of these bits is 0x0114. Accumulated energy = (SYS_ENM_H_DDR1:SYS_ENM_L_DDR1)/6174020000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.
[31:0]	SYS_ENM_L_DDR1	RO	Least significant 32 bits of a 64-bit representation of the accumulated energy consumption of DDR 1: <ul style="list-style-type: none"> The memory address offset of these bits is 0x0140. Accumulated energy = (SYS_ENM_H_DDR1:SYS_ENM_L_DDR1)/6174020000 joules. The CB_nRST reset signal resets the register to zero. The register then updates every 100µs after the reset.

5.11 UART memory addresses and control registers

The Morello SoC and IOFPGA contain registers that control the UARTs in the Morello System Development Platform.

The following table shows the Morello SDP UART memory addresses.

Table 5-213: UART memory locations

UART	Memory address	Comment
APUART0	0x00_2A40_0000	AP peripherals memory map. See Application Processor subsystem peripherals memory map .
APUART1	0x00_2A41_0000	AP peripherals memory. See Application Processor subsystem peripherals memory map .
SCPUART	0x00_4400_2000	SCP peripherals memory map. See System Control Processor peripherals memory map .
MCPUART0	0x00_4C00_2000	MCP peripherals memory map. See Manageability Control Processor peripherals memory map .
MCPUART1	0x00_4400_3000	MCP peripherals memory map. See Manageability Control Processor peripherals memory map .
FPGAUART1	0x00_1C09_0000	IOFPGA memory map. See IOFPGA memory map .
FPGAUART2	0x00_1C0A_0000	IOFPGA memory map. See IOFPGA memory map .



APUART1 is used to communicate with the MCP through MCPUART1 and is accessible to the Application Processor (AP) cores. MCPUART1 is not accessible to the AP cores.

The following table, from the PL011 Technical Reference Manual, shows the UART0 and UART1 control registers in address offset order from the base memory address. In the table, UART0 and UART1 refer to:

- APUART0 and APUART1 respectively.
- MCPUART0 and MCPUART1 respectively.
- FPGAUART1 and FPGAUART2 respectively.

UART0 refers to SCPUART.

Undefined registers are reserved. Software must not attempt to access these registers.

Table 5-214: UART control registers summary

Offset	Name	Type	Reset value	Width	Function
0x0000	UARTODR	RW	-	32	Data Register.
0x0004	UARTORSR/UARTOECR	RW	0x0000_0000	32	Receive Status Register/Error Clear Register.
0x0018	UARTOFR	RO	0x0000_0012	32	Flag Register.
0x0020	UARTOILPR	RW	0x0000_0000	32	IrDA Low-Power Counter Register.
0x0024	UARTOIBRD	RW	0x0000_0000	32	Integer Baud Rate Register.
0x0028	UARTOFBRD	RW	0x0000_0000	32	Fractional Baud Rate Register.
0x002C	UARTOLCR_H	RW	0x0000_0000	32	Line Control Register.
0x0030	UARTOCR	RW	0x0000_0300	32	Control Register.
0x0034	UARTOIFLS	RW	0x0000_0012	32	Interrupt FIFO Level Select Register.
0x0038	UARTOIMSC	RW	0x0000_0000	32	Interrupt Mask Set/Clear Register.
0x003C	UARTORIS	RO	0x0000_0000	32	Raw Interrupt Status Register.
0x0040	UARTOMIS	RO	0x0000_0000	32	Masked Interrupt Status Register.
0x0044	UARTOICR	WO	-	32	Interrupt Clear Register.
0x0048	UARTODMACR	RW	0x0000_0000	32	DMA Control Register.

Offset	Name	Type	Reset value	Width	Function
0x0FE0	UART0PeriphID0	RO	0x0000_0011	32	UART0 peripheral ID Register 0.
0x0FE4	UART0PeriphID1	RO	0x0000_0010	32	UART0 peripheral ID Register 1.
0x0FE8	UART0PeriphID2	RO	0x0000_0004	32	UART0 peripheral ID Register 2.
0x0FEC	UART0PeriphID3	RO	0x0000_0000	32	UART0 peripheral ID Register 3.
0x0FF0	UART0PCellID0	RO	0x0000_000D	32	UART0 component ID Register 0.
0x0FF4	UART0PCellID1	RO	0x0000_00F0	32	UART0 component ID Register 1.
0x0FF8	UART0PCellID2	RO	0x0000_0005	32	UART0 component ID Register 2.
0x0FFC	UART0PCellID3	RO	0x0000_00B1	32	UART0 component ID Register 3.
0x1000	UART1DR	RW	-	32	Data Register.
0x1004	UART1RSR/UART1ECR	RW	0x0000_0000	32	Receive Status Register/Error Clear Register.
0x1018	UART1FR	RO	0x0000_0012	32	Flag Register.
0x1020	UART1ILPR	RW	0x0000_0000	32	IrDA Low Power Counter Register.
0x1024	UART1IBRD	RW	0x0000_0000	32	Integer Baud Rate Register.
0x1028	UART1FBRD	RW	0x0000_0000	32	Fractional Baud Rate Register.
0x102C	UART1LCR_H	RW	0x0000_0000	32	Line Control Register.
0x1030	UART1CR	RW	0x0000_0300	32	Control Register.
0x1034	UART1IFLS	RW	0x0000_0012	32	Interrupt FIFO Level Select Register.
0x1038	UART1IMSC	RW	0x0000_0000	32	Interrupt Mask Set/Clear Register.
0x103C	UART1RIS	RO	0x0000_0000	32	Raw Interrupt Status Register.
0x1040	UART1MIS	RO	0x0000_0000	32	Masked Interrupt Status Register.
0x1044	UART1ICR	WO	-	32	Interrupt Clear Register.
0x1048	UART1DMACR	RW	0x0000_0000	32	DMA Control Register.
0x1FE0	UART1PeriphID0	RO	0x0000_0011	32	UART1 peripheral ID Register 0.
0x1FE4	UART1PeriphID1	RO	0x0000_0010	32	UART1 peripheral ID Register 1.
0x1FE8	UART1PeriphID2	RO	0x0000_0004	32	UART1 peripheral ID Register 2.
0x1FEC	UART1PeriphID3	RO	0x0000_0000	32	UART1 peripheral ID Register 3.
0x1FF0	UART1PCellID0	RO	0x0000_000D	32	UART1 component ID Register 0.
0x1FF4	UART1PCellID1	RO	0x0000_00F0	32	UART1 component ID Register 1.
0x1FF8	UART1PCellID2	RO	0x0000_0005	32	UART1 component ID Register 2.
0x1FFC	UART1PCellID3	RO	0x0000_00B1	32	UART1 component ID Register 3.

See the *Arm® PrimeCell UART(PL011) Technical Reference Manual* for more information.

Appendix A Signal descriptions

This appendix describes the signals that are present at the Morello SDP ports.

A.1 Morello-SoC JTAG connector

There is one 20-pin JTAG box header connector on the back panel.

The I/O voltage of the JTAG connector is 1V8.

The following table shows the pin mapping of the P-JTAG connector.

Table A-1: Morello SoC P-JTAG connector signal list

Pin	Signal	Pin	Signal
1	VTREF	2	No connection
3	nTRST	4	GND
5	TDI	6	GND
7	TMS	8	GND
9	TCK	10	GND
11	RTCK	12	GND
13	TDO	14	GND
15	nSRSTI	16	GND
17	DBGREQ	18	GND
19	DBGACK	20	GND



Note

- Pin 1 is the lower left pin of the connector. Pin 2 is the upper left pin of the connector.
- Pins 1, 5, 7, 13, 15, and 19 have pullup resistors to 1V8.
- Pins 3, 9, 11, and 17 have pulldown resistors to GND.

Related information

[The Morello SDP at a glance](#) on page 19

A.2 Trace connector

There is a Samtec QSH 60-pin plug connector on the back panel which supports 32-bit trace, JTAG debug, and Serial Wire Debug (SWD).

The I/O voltage for the connector is 1V8.

The following table shows the pin mapping of the trace connector.

Table A-2: Trace connector pin mapping

Pin	Signal	Pin	Signal
1	DEBUG_VTREF	2	SOC_TMS
3	SOC_TCK	4	SOC_TDO
5	SOC_TDI	6	CS_nTRSTI
7	SOC_RTCK	8	SOC_nTRST
9	SOC_nTRST	10	DBGRRQ
11	DBGACK	12	TRACE_VTREF
13	TRACE_CLKA	14	TRACE_CLKB
15	GND	16	GND
17	TRACE_CTL	18	TRACE_DATA19
19	TRACE_DATA0	20	TRACE_DATA20
21	TRACE_DATA1	22	TRACE_DATA21
23	TRACE_DATA2	24	TRACE_DATA22
25	TRACE_DATA3	26	TRACE_DATA23
27	TRACE_DATA4	28	TRACE_DATA24
29	TRACE_DATA5	30	TRACE_DATA25
31	TRACE_DATA6	32	TRACE_DATA26
33	TRACE_DATA7	34	TRACE_DATA27
35	TRACE_DATA8	36	TRACE_DATA28
37	TRACE_DATA9	38	TRACE_DATA29
39	TRACE_DATA10	40	TRACE_DATA30
41	TRACE_DATA11	42	TRACE_DATA31
43	TRACE_DATA12	44	No connection
45	TRACE_DATA13	46	No connection
47	TRACE_DATA14	48	No connection
49	TRACE_DATA15	50	No connection
51	TRACE_DATA16	52	No connection
53	TRACE_DATA17	54	No connection
55	TRACE_DATA18	56	No connection
57	GND	58	GND
59	No connection	60	No connection



Pin 17, TRACE_CTL, is not used and has a pulldown resistor to GND.

Related information

[The Morello SDP at a glance](#) on page 19

A.3 Front panel I/O header

There is a 20-pin header, 10×2, on the Morello board near the front panel. The header provides connectivity for LEDs and switches between the front panel and the board.

Some signals are not brought out to the front panel but are available on the header but are available for use at the connector. The following table shows the pin mapping of the front panel I/O header.

Table A-3: Front panel I/O header pin mapping

Pin	Polarity	Pin	Polarity	Description
1	+	2	-	Connects PBON button on board to PBON button on front panel.
3	+	4	-	Connects PBRESET button on board to PBRESET button on front panel.
5	+	6	-	Reserved
7	-	8	+	Connects to orange LED in 4-level light pipe, fourth from bottom, on back panel. Denotes MCC USB activity. These pins are not brought out to the front panel.
9	-	10	+	Connects to green LED in 4-level light pipe, third from bottom, on back panel. Denotes MCC USB activity. These pins are not brought out to the front panel.
11	-	12	+	Power LED embedded in PBON button on front panel.
13	-	14	+	HDD activity LED on front panel. This signal is a combined signal from SATA0 and SATA1.
15	-	16	+	Connects to the GbE activity LED embedded in the GbE connector and denotes GbE traffic. These pins are not brought out to the front panel.
17	-	18	+	Connects to blue LED in 4-level light pipe, first from bottom. Denotes UID activity. These pins are not brought out to the front panel.
19	N/A	20	N/A	No connection

Related information

[The Morello SDP at a glance](#) on page 19

A.4 PCI Express and CCIX slots

There are two 16-lane PCIe slots, one 4-lane PCIe slot, and one dual-use 16-lane PCIe-Cache-Coherent Interconnect for Accelerators (CCIX) slot on the Morello board.

The following table shows the PCIe slots and the number of lanes implemented.

Table A-4: PCI Express expansion slots

Slot number	PCIe lane connector size	Used lanes	Unused lanes	Comment
Slot 1	×4	1	3	PCIe
Slot 2	×16	16	0	PCIe
Slot 3	×16	8	8	PCIe
Slot 4	×16	16	0	PCIe CCIX dual-use

Related information

[The Morello SDP at a glance](#) on page 19

A.5 Power connectors

There are an ATX 24-pin power connector and an ATX/EPS 8-pin secondary connector on the Morello board.

The ATX 24-pin power connector has the standard ATXv2.2 pin connections.

The following table shows the ATX/EPS connector pin mapping.

Table A-5: ATX/EPS connector pin mapping

Pin	Connection
1	GND
2	GND
3	GND
4	GND
5	12V
6	12V
7	12V
8	12V

Related information

[The Morello SDP at a glance](#) on page 19

Appendix B Rainier clusters

This appendix is a short overview of the Rainier cluster and describes the CPU ID, to be used in conjunction with the *Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture*.

B.1 About Rainier

The Rainier cluster is a high-performance and low-power Arm prototype dual-core element that implements the Armv8-A architecture.

The Rainier cluster supports:

- The Armv8.2-A extension.
- The RAS extension.
- The Statistical Profiling extension.
- ARMv8.3-RCpc, Weaker release consistency: three Load-Acquire RCpc Register (LDAPR) instructions are introduced by the Armv8.3-A extension.
- The Load acquire (LDAPR) instructions introduced in the Armv8.3-A extension.
- ARMv8.0-SSBS, Speculative Store Bypass Safe bit: The PSTATE.SSBS bit is added to allow software to indicate whether hardware is permitted to load or store speculatively in a manner that could give rise to a cache timing side channel, which in turn could be used to derive an address from values loaded to a register from memory.
- ARMv8.2-DotProd, SIMD Dot Product: The SIMD dot product instructions are added to the A64 and A32/T32 instruction sets.



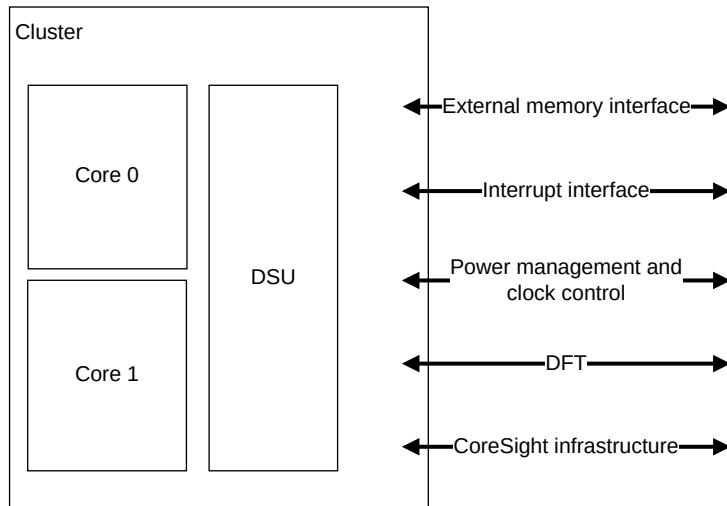
Note

The Morello architecture is supported in AArch64 state only. Morello introduces a new data type to Arm®v8-A architecture profile, Capability, an unforgeable token of authority that provides a foundation for fine-grained memory protection and strong compartmentalization. The architecture extends the Arm®v8 AArch64 state with the principles proposed in version 8 of the Capability Hardware Enhanced RISC Instructions: CHERI Instruction-Set Architecture, to provide hardware support for fine-grained protection, and building blocks for secure, scalable compartmentalization.

For a full description of the Armv8-A architecture capability profile, read the *Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture*.

The following figure shows the Rainier cluster.

Figure B-1: Rainier cluster



Rainier has a Level 1 (L1) memory system and a private, integrated Level 2 (L2) cache. It also includes a superscalar, variable-length, out-of-order pipeline.

Rainier is implemented inside the DynamIQ Shared Unit (DSU) cluster.

B.2 Rainier features and components

The Rainier clusters in the Morello SoC contain the following features and components.

Major components of the Rainier clusters in the Morello SoC

The Morello SoC contains two dual-core Rainier clusters. Each cluster has:

- 64KB private L1 instruction cache for each core.
- 64KB private L1 data cache for each core.
- 1MB private L2 unified cache for each core.
- 1MB shared L3 unified cache in the DynamIQ Shared Unit (DSU) Flash Cache Module (FCM).
- Digital Storage Oscilloscope (DSO) for voltage sensing and logic monitoring, with current stimulus generated by a separate Noise Generator (NG).

Core features

The major features of the Rainier cores are:

- 48-bit Physical Address (PA).
- Memory Management Unit (MMU).
- Cryptographic Extension.

- Superscalar, variable-length, out-of-order pipeline.
- Support for Arm TrustZone® technology.
- Reliability, Availability, and Serviceability (RAS) Extension.
- Generic Interrupt Controller (GICv4) CPU interface to connect to an external distributor.
- Generic Timers interface supporting 64-bit count input from an external system counter.
- Integrated execution unit that implements the Advanced SIMD and floating-point architecture support.
- AArch64 Execution state at all Exception levels (EL0 to EL3).

Cache features

The Rainier cache features are:

- Separate L1 data and instruction caches.
- Private, unified data and instruction L2 cache.
- L1 and L2 memory protection in the form of Error Correcting Code (ECC) or parity on RAM instances which affect functionality.
- Configurable instruction cache hardware coherency.

Debug features

The Rainier debug features are:

- Armv8.2 debug logic.
- Performance Monitoring Unit (PMU).
- Statistical Profiling Extension (SPE).
- CoreSight Embedded Logic Analyzer (ELA).
- Embedded Trace Macrocell (ETM) that supports instruction trace only.

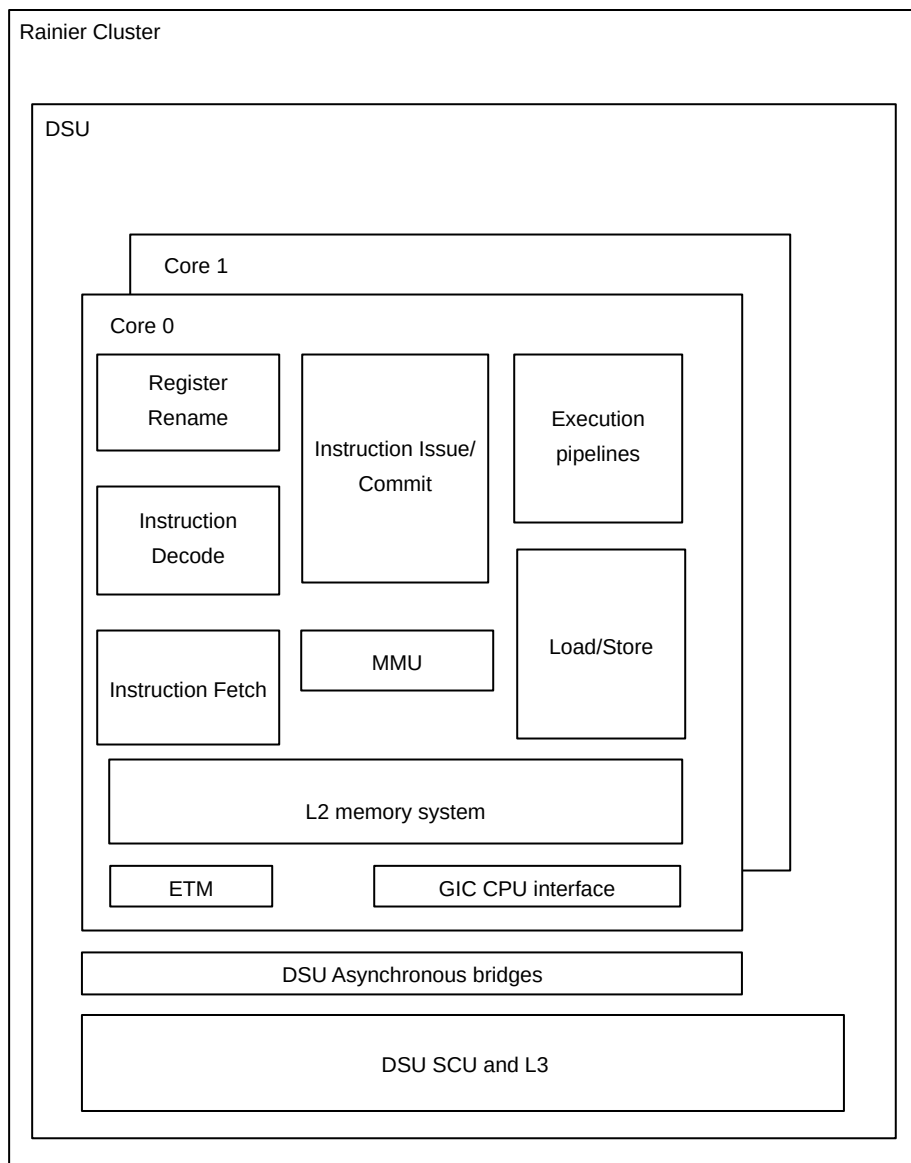
Rainier core components

The main components of the Rainier core are:

- Instruction fetch
- Instruction decode
- Register rename
- Instruction issue
- Execution pipelines
- L1 data memory system
- L2 memory system

The following figure is an overview of the Rainier cores.

Figure B-2: Rainier core overview



Instruction fetch

The instruction fetch unit fetches instructions from the L1 instruction cache and delivers the instruction stream to the instruction decode unit.

The instruction fetch unit includes:

- 64KB, 4-way, set associative L1 instruction cache with 64-byte cache lines and parity protection.

- Fully associative L1 instruction TLB with 64KB page sizes.
- Dynamic branch predictor.
- Configurable support for instruction cache hardware coherency.

Instruction decode

The instruction decode unit supports the A64 instruction set. It also supports Advanced SIMD and floating-point instructions.

Register rename

The register rename unit performs register renaming to facilitate out-of-order execution and dispatches decoded instructions to various issue queues.

Instruction issue

The instruction issue unit controls when the decoded instructions are dispatched to the execution pipelines. It includes issue queues for storing instruction pending dispatch to execution pipelines.

Execution pipeline

The execution pipeline includes:

- Integer execute unit that performs arithmetic and logical data processing operations.
- Vector execute unit that performs Advanced SIMD and floating-point operations. It can execute the cryptographic instructions.

L1 data memory system

The L1 data memory system executes load and store instructions and encompasses the L1 data side memory system. It also services memory coherency requests.

The load/store unit includes:

- 64KB, 4-way, set associative L1 data cache with 64-byte cache lines and ECC protection per 32 bits.
- Fully associative L1 data TLB with 64KB page sizes.

L2 memory system

The L2 memory system services L1 instruction and data cache misses in the Rainier core.

The L2 memory system includes:

- 8-way set associative L2 cache with data ECC protection per 64 bits. The L2 cache is 1024KB.
- Interface with the DynamIQ Shared Unit (DSU) configured for asynchronous operation.

B.3 Main ID Register, MIDR_EL1

The MIDR_EL1 provides identification information for the Rainier core, including an implementer code for the device and a device ID number.



For general Rainier programming information, see *Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture*.

Bit field descriptions

MIDR_EL1 is a 32-bit register. This register is read-only. The full register reset value is `0x3F0F_4120`.

Figure B-3: Rainier MIDR_EL1 bit assignments

31	24	23	20	19	16	15	8	7	4	3	0
Implementer				Variant		Architecture		PartNum		HW	Revision

Implementer, [31:24]

Indicates the implementer code. This code value is `0x3F` - University/research.

Variant, [23:20]

Indicates the variant number of the core. This number is the major revision number *x* in the *rx* part of the *rxpy* description of the product revision status. This value is `0x0` - revision 0.

Architecture, [19:16]

Indicates the architecture code. This value is `0x0F` - CPUID scheme.

PartNum, [15:8]

Indicates the primary part number. This value is `0x41` - Rainier core (Morello).

HW, [7:4]

Indicates hardware number. This value is `0x2` - Morello SoC.

Revision, [3:0]

Indicates the minor revision number of the core. This is the minor revision number *y* in the *py* part of the *rxpy* description of the product revision status. This value is `0x0` - part 0.

Configuration

The MIDR_EL1 is architecturally mapped. Bit fields and details are defined in the *Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture*.

Appendix C CMN-Skeena

This appendix is a short overview of the CMN-Skeena coherent interconnect, within Morello.

C.1 About CMN-Skeena coherent mesh network

CMN-Skeena is a prototype coherent interconnect designed to enable the transportation of tags from Rainier cores to DMC-Bing.

CMN-Skeena, the primary interconnect in Morello, is based on the Arm® CoreLink™ CMN-600 Coherent Mesh Network product, itself a configurable coherent interconnect designed to meet the Power, Performance, and Area (PPA) requirements for coherent mesh network systems that are used in high-end networking and enterprise compute applications. The Morello platform implements the CHERI Capability extension to the Arm architecture in the form of tagged bits. The modifications to CMN-600 that have resulted in CMN-Skeena include:

- An additional 129th bit to data path.
- An additional transport bit in existing packets, known as a “Poison” bit.
- An updated SLC: the poison status is stored as special encoding of ECC in RAMs.
- Poison generation/handling is disabled to ensure propagation of bit.

For more information on CMN-600, see the *Arm® CoreLink™ CMN-600 Coherent Mesh Network Technical Reference Manual*. For more information on the capability architecture, see the *Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture*.

Compliance

CMN-Skeena is based on the AMBA 5 CHI Issue E architecture specification, and implements the following capabilities:

- Fully compliant with CHI interconnect architecture.
- Non-blocking coherence protocol.
- Packet-based communication.
- The following four types of channels:
 - Request (REQ).
 - Response (RSP).
 - Snoop (SNP).
 - Data (DAT).
- Credited end-to-end protocol-layer flow-control with a retry-once mechanism for flexible bandwidth and resource allocation.
- Integrated end-to-end Quality-of-Service (QoS) capabilities.

See the *AMBA® 5 CHI Architecture Specification* for more information.

CMN-Skeena features

CMN-Skeena provides the following features:

- 5 x 2 mesh network topology.
- Two Fully coherent Requesting Node (RN-F) interfaces to CHI.E-based Rainier compute clusters.
- Two RN-F interfaces to CHI.E-based DMC-Bing memory controllers.
- 4MB system cache size.
- 8MB Snoop Filter (SF)
- I/O Home Node (HN-I) for a direct connection to CCIX AXI-subordinate interface to enable low latency pathways.
- Data channel: a pair of 256-bit data channels, one for each direction.
- DVM message transport between managers.
- QoS regulation for shaping traffic profiles.
- A Performance Monitoring Unit (PMU) to count performance-related events.
- RAS features including transport parity, optional data path parity, and SECDED ECC.

CMN-Skeena and Morello SoC connections

CMN-Skeena contains the following protocol nodes and devices:

RN-F (Fully coherent Requesting Node)

A fully coherent manager device that supports CHI.E RN-F

RN-D (I/O coherent Requesting Node with DVM support) bridge

An I/O coherent manager device that supports accepting DVM messages on the snoop channel.

HN-F (Fully coherent Home Node)

A device that acts as a home node for a coherent region of memory, accepting coherent requests from RN-Fs and RN-Is, and generating snoops to all applicable RN-Fs in the system as required to support the coherency protocol.

HN-I (I/O Home Node)

A device that acts as a home-node for the subordinate I/O subsystem, responsible for ensuring proper ordering of requests targeting the subordinate I/O subsystem. HN-I supports AMBA AXI or ACE-Lite.

HN-D (HN-I + DVM Node)

A device that includes HN-I, DVM Node (DN), Configuration Node (CFG), Global Configuration Subordinate, and the Power/Clock Control Block (PCCB).

SN-F (CHI Slave Node)

A device which solely is a recipient of CHI commands, limited to fulfilling simple read, write, and CMO requests targeting normal memory.

SBSX bridge

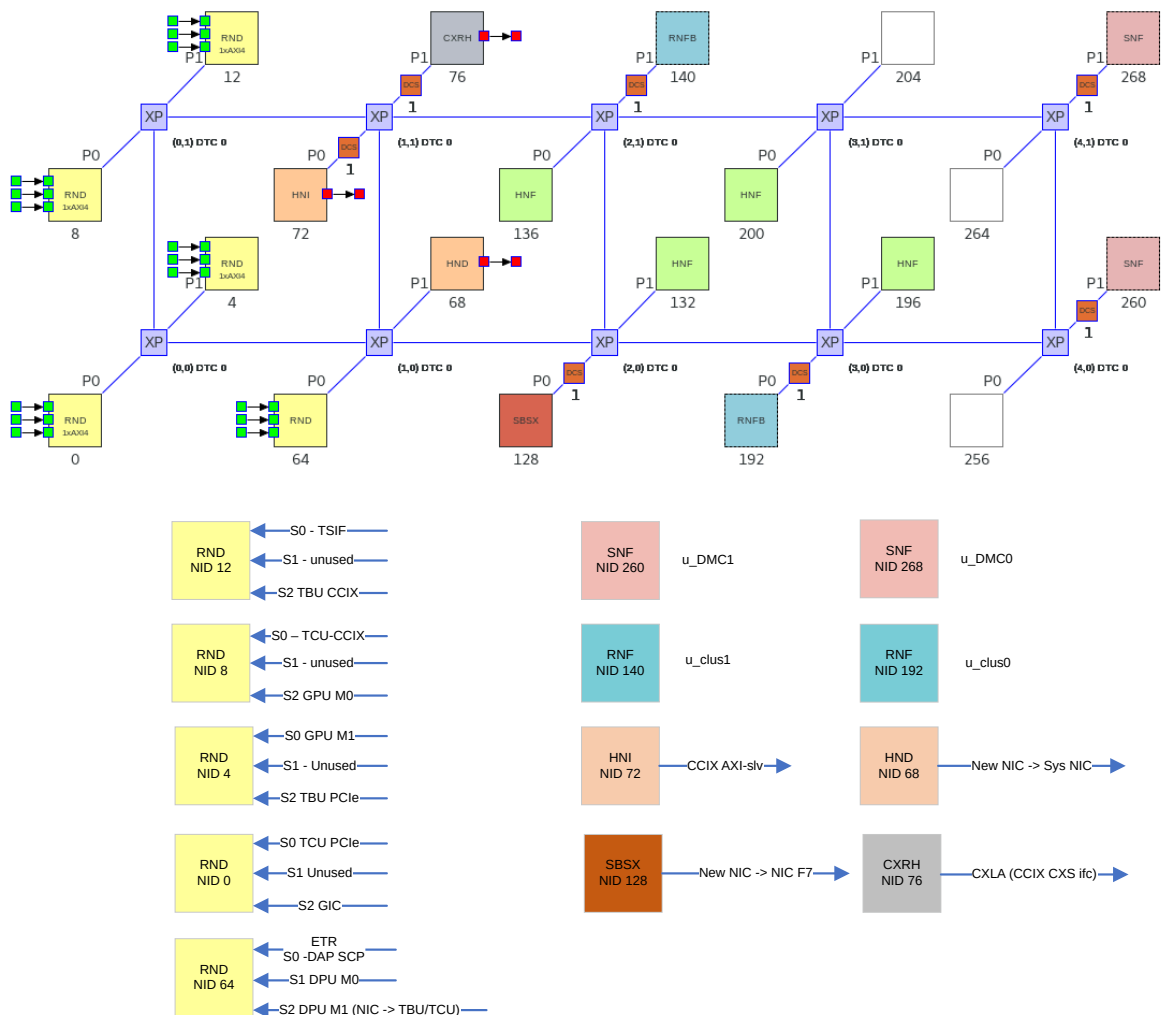
A CHI bridge device that converts simple CHI read, write, and CMO commands to an ACE-Lite subordinate memory device.

CXG bridge

A CXG device bridges between CHI and CXS (CCIX port) and contains: CCIX Request Agent (CXRA) proxy and CCIX Home Agent (CXHA) proxy functionality; CXS Link Agent (CXLA) functionality which is external to the CMN-600 hierarchy.

The following figure shows the CMN-Skeena mesh diagram, listing external SoC connections.

Figure C-1: CMN-Skeena mesh and SoC connections



Appendix D DMC-Bing

This appendix is a short overview of DMC-Bing dynamic memory controller. It includes new DMC-Bing system registers and a brief description of the Client mode and Server mode.

D.1 About the Bing dynamic memory controller

The Bing memory controller is prototype dynamic memory controller developed specifically to support the Morello prototype capability architecture and for use on the Morello board. It is an adaptation of the existing Arm® CoreLink DMC-620 Dynamic Memory Controller to support the capability tags that are defined in the Morello architecture. The Morello board is designed to incorporate two Rainier CPU cores and two Bing DMCs. For more information, see the *Arm® CoreLink™ DMC-620 Dynamic Memory Controller Technical Reference Manual*.

Bing operates in two distinct modes: Client Mode and Server Mode. The principal difference is that Server Mode assumes that ECC DRAM is installed. Client Mode does not make that assumption. It supports a usage model associated with client mobile devices, although it uses ECC capabilities if ECC DRAM is installed.

The Morello capability tag is a single bit attached to a 128-bit word in memory. To support this with commonly available DRAMs, there are two separate mechanisms that are used depending on whether Bing is operating in Client Mode or Server Mode.

Server Mode

Server Mode can be used if ECC DRAM is installed. The capability tag bit is stored in one of the ECC bits. However, the way that capability tags are implemented using one of the ECC bits forces the ECC function to be implemented across 128 bits rather than 64. A single-bit error can be corrected, and a double-bit error is detected across 128 bits. The double-bit error is reported as a Non-Data-Error (NDERR) in the recommended setup.

Client Mode

In Client Mode, no assumption about the existence of ECC DRAM is made. The tag bits are stored in DRAM in a memory carve-out region. This region consumes 1/128 of the available total system memory, at the top of the installed physical address space, and renders that amount of memory unusable for other system functions. There is a performance penalty in having to go out to DRAM to read the tags, so Bing includes a multi-level internal cache scheme to improve performance. Only the cast out tags are written to, and read from, memory.

The internal tag caches are invalidated at reset of the DMC. The reset sequence must include zeroing of the entire carve-out region in system DRAM.

If ECC DRAM is installed, then the same 128-bit ECC scheme as for Server Mode is implemented. However, there are special considerations because a memory transaction in Client Mode can involve up to two memory accesses: one to data space (if a data cache miss occurs); and one to the carve-out region (if a tag cache miss occurs). ECC errors for accesses to data space occur as

previously described. However, ECC errors on read and write misses to the carve-out region are treated differently.

An ECC error arising from a read miss to the carve-out region is handled as follows:

- Single-bit errors are corrected and the result is placed in a tag cache as usual.
- Double-bit errors are detected and the indication is merged with the read response of the data access. The data caches and the tag caches are not updated, in this case.

An ECC error arising from a write miss to the carve-out region is handled as follows:

- Single-bit errors are corrected and the result is placed in a tag cache as usual.
- Double-bit errors are detected but no special action is taken. The data caches and the tag caches are updated.

D.2 Bing system registers

The following table describes the differences introduced into the DMC-620 configuration by Bing. Use this table as a supplement to the *Arm® CoreLink™ DMC-620 Dynamic Memory Controller Technical Reference Manual*. It includes information only that supersedes the DMC-620 TRM.

Table D-1: DMC-620 System register changes for DMC-Bing

Offset	Name	Type	Reset Value	Width	Description	Change from DMC-620
0x538	failed_access_int_/info_31_00	RO	0	1	Bit 27. 1: failed_access_int_info_/ client_carve_out_fail	New bit field added.
0x708	errOctrl_de	-	0	-	0 disable	Reset value change.
-	errOctrl_ecc	-	2'b01	-	2'b01 SECDEC ecc enabled	Reset value change.
-	errOctrl_wb	-	0	-	0 disable	Reset value change.
-	errOctrl_retry	-	0	-	0 disable	Reset value change.
-	errOctrl_rmw	-	1	-	1 enable	Reset value change.
-	errOctrl_wde	-	0	-	0 disable	Reset value change.
-	memory_width	2'b11	0	-	2'b11 128-bit	Reset value change.
-	scrub_enable0	-	0	-	0 disable	Reset value change.
-	scrub_enable1	-	0	-	0 disable	Reset value change.
-	sram_bitflip_/ inject0	-	0	-	0 disable	Reset value change.
-	sram_bitflip_inject	-	0	-	0 disable	Reset value change.
-	errOctrl_derr_as/ _nderr	RW	1	1	Bit 10 0: Double data errors resp=2'b00 (NORMALOKAY) 1: Double data errors resp=2'b11 (NONDATAERR)	New bit field added.

In addition to the existing DMC-620 registers, DMC-Bing has six new system registers.

Table D-2: Bing system registers

Offset	Name	Type	Reset	Width	Description
0xD00	capability_ctl	RW	0	1	0+: 1 capability_mode: - 0 = server mode - 1 = client mode
0xD04	tag_cache_ctl	WO	0	1	<ul style="list-style-type: none"> 0+: 1 invalidate_l1_tag_cache
0xD08	tag_cache_cfg	RW	3'b111	3	<ul style="list-style-type: none"> 0+: 1 l1_tag_cache_en 1+: 1 c1_tag_cache_en 2+: 1 c2_tag_cache_en
0xD0C	memory_access_ctl	RW	0_0f0f	12	<ul style="list-style-type: none"> 0+: 4 read_qos 8+: 4 write_qos 16+: 1 memory_access checking disable
0xD10	memory_address_ctl	RW	0	32	<ul style="list-style-type: none"> 0+: 32 carve_out_base_address_31_0
0xD14	memory_address_ctl2	RW	0	32	<ul style="list-style-type: none"> 0+: 32 carve_out_base_address_64_32
0xD18	bing_special_ctrl_reg	RW	0	32	<ul style="list-style-type: none"> 0+: 32 bing_special_ctrl

Server Mode setup

Bing starts in Server Mode at reset. No additional configuration is necessary to set up Bing to operate in Server Mode.

Client Mode setup

If Client Mode operation is wanted, there are three steps that must be taken:

1. The following values should be programmed into the previous registers to set up the tested memory configuration of 8GB (4GB per channel) and carve-out region of 1/128 of the total memory at the top of the address space.

Register	Offset	Value
memory_access_ctl	0xd0c	0x0001_0f0f
carve_out_base_address_31_0	0xd10	0xff00_0000
carve_out_base_address_64_32	0xd14	0x0000_0000
memory_address_max_31_00	0x078	0xfdf_0010
memory_address_max_47_32	0x07c	0x0000_0000

2. The entire carve-out region in system DRAM (1/128 of the total memory, at the top of the address space) must be written to 0.
3. The following registers must be programmed as specified:

Offset	Name	Type	Value at Reset	Width	Must be set as described
0x130	Feature_config.si_clk_gate_disable	-	Bit 8 = 0	-	Bit 8 = 1 disable si clk gate
0xd00	capability_ctl	-	Bit 0 = 0	-	Bit 0 = 1 to enable Client Mode

PMU counters

There are many new PMU counters in Bing. These PMU counters are described in the following table.

Table D-5: Bing PMU counters

Offset	Name	Type	Reset	Width	Description
0xba0	pmu_clk_counter_2_mask_31_00	RW	0	32	0+: 32 pmu_clk_counter_2_mask_31_00
0xba4	pmu_clk_counter_2_mask_63_32	RW	0	12	0+= 12 pmu_clk_counter_2_mask_63_32
0xba8	pmu_clk_counter_2_match_31_00	RW	0	32	0+= 32 pmu_clk_counter_2_match_31_00
0xbac	pmu_clk_counter_2_match_63_32	RW	0	12	0+= 12 pmu_clk_counter_2_match_63_32
0xbb0	pmu_clk_counter_2_control	RW	0	9	<ul style="list-style-type: none"> 0+= 1 pmu_clk_counter_2_enable 1+= 1 pmu_clk_counter_2_invert_match 2+= 5 pmu_clk_counter_2_event_mux 7+= 2 pmu_clk_counter_2_incr_mux
0xbb8	pmu_clk_counter_2_snapshot_value_31_00	RO	0	32	0+= 32 pmu_clk_counter_2_snapshot_value_31_00
0xbc0	pmu_clk_counter_2_counter_value_31_00	RW	0	32	0+= 32 pmu_clk_counter_2_counter_value_31+00
0xbc8	pmu_clk_counter_3_mask_31_00	RW	0	32	0+: 32 pmu_clk_counter_3_mask_31_00
0xbcc	pmu_clk_counter_3_mask_63_32	RW	0	12	0+= 12 pmu_clk_counter_3_mask_63_32
0xbd0	pmu_clk_counter_3_match_31_00	RW	0	32	0+= 32 pmu_clk_counter_3_match_31_00
0xbd4	pmu_clk_counter_3_match_63_32	RW	0	12	0+= 12 pmu_clk_counter_3_match_63_32
0xbd8	pmu_clk_counter_3_control	RW	0	9	<ul style="list-style-type: none"> 0+= 1 pmu_clk_counter_3_enable 1+= 1 pmu_clk_counter_3_invert_match 2+= 5 pmu_clk_counter_3_event_mux 7+= 2 pmu_clk_counter_3_incr_mux
0xbe0	pmu_clk_counter_3_snapshot_value_31_00	RO	0	32	0+= 32 pmu_clk_counter_3_snapshot_value_31_00
0xbe8	pmu_clk_counter_3_counter_value_31_00	RW	0	32	0+= 32 pmu_clk_counter_3_counter_value_31+00
0xbf0	pmu_clk_counter_4_mask_31_00	RW	0	32	0+: 32 pmu_clk_counter_4_mask_31_00
0xbf4	pmu_clk_counter_4_mask_63_32	RW	0	12	0+= 12 pmu_clk_counter_4_mask_63_32
0xbf8	pmu_clk_counter_4_match_31_00	RW	0	32	0+= 32 pmu_clk_counter_4_match_31_00
0xbfc	pmu_clk_counter_4_match_63_32	RW	0	12	0+= 12 pmu_clk_counter_4_match_63_32
0xc00	pmu_clk_counter_4_control	RW	0	9	<ul style="list-style-type: none"> 0+= 1 pmu_clk_counter_4_enable 1+= 1 pmu_clk_counter_4_invert_match 2+= 5 pmu_clk_counter_4_event_mux 7+= 2 pmu_clk_counter_4_incr_mux
0xc08	pmu_clk_counter_4_snapshot_value_31_00	RO	0	32	0+= 32 pmu_clk_counter_4_snapshot_value_31_00
0xc10	pmu_clk_counter_4_counter_value_31_00	RW	0	32	0+= 32 pmu_clk_counter_4_counter_value_31+00
0xc18	pmu_clk_counter_5_mask_31_00	RW	0	32	0+: 32 pmu_clk_counter_5_mask_31_00
0xc1c	pmu_clk_counter_5_mask_63_32	RW	0	12	0+= 12 pmu_clk_counter_5_mask_63_32
0xc20	pmu_clk_counter_5_match_31_00	RW	0	32	0+= 32 pmu_clk_counter_5_match_31_00
0xc24	pmu_clk_counter_5_match_63_32	RW	0	12	0+= 12 pmu_clk_counter_5_match_63_32

Offset	Name	Type	Reset	Width	Description
0xc28	pmu_clk_counter_5_control	RW	0	9	<ul style="list-style-type: none"> 0 += 1 pmu_clk_counter_5_enable 1 += 1 pmu_clk_counter_5_invert_match 2 += 5 pmu_clk_counter_5_event_mux 7 += 2 pmu_clk_counter_5_incr_mux
0xc30	pmu_clk_counter_5_snapshot_value_31_00	RO	0	32	0 += 32 pmu_clk_counter_5_snapshot_value_31_00
0xc38	pmu_clk_counter_5_counter_value_31_00	RW	0	32	0 += 32 pmu_clk_counter_5_counter_value_31+00
0xc40	pmu_clk_counter_6_mask_31_00	RW	0	32	0 += 32 pmu_clk_counter_6_mask_31_00
0xc44	pmu_clk_counter_6_mask_63_32	RW	0	12	0 += 12 pmu_clk_counter_6_mask_63_32
0xc48	pmu_clk_counter_6_match_31_00	RW	0	32	0 += 32 pmu_clk_counter_6_match_31_00
0xc4c	pmu_clk_counter_6_match_63_32	RW	0	12	0 += 12 pmu_clk_counter_6_match_63_32
0xc50	pmu_clk_counter_6_control	RW	0	9	<ul style="list-style-type: none"> 0 += 1 pmu_clk_counter_6_enable 1 += 1 pmu_clk_counter_6_invert_match 2 += 5 pmu_clk_counter_6_event_mux 7 += 2 pmu_clk_counter_6_incr_mux
0xc58	pmu_clk_counter_6_snapshot_value_31_00	RO	0	32	0 += 32 pmu_clk_counter_6_snapshot_value_31_00
0xc60	pmu_clk_counter_6_counter_value_31_00	RW	0	32	0 += 32 pmu_clk_counter_6_counter_value_31+00
0xc68	pmu_clk_counter_7_mask_31_00	RW	0	32	0 += 32 pmu_clk_counter_7_mask_31_00
0xc6c	pmu_clk_counter_7_mask_63_32	RW	0	12	0 += 12 pmu_clk_counter_7_mask_63_32
0xc70	pmu_clk_counter_7_match_31_00	RW	0	32	0 += 32 pmu_clk_counter_7_match_31_00
0xc74	pmu_clk_counter_7_match_63_32	RW	0	12	0 += 12 pmu_clk_counter_7_match_63_32
0xc78	pmu_clk_counter_7_control	RW	0	9	<ul style="list-style-type: none"> 0 += 1 pmu_clk_counter_7_enable 1 += 1 pmu_clk_counter_7_invert_match 2 += 5 pmu_clk_counter_7_event_mux 7 += 2 pmu_clk_counter_7_incr_mux
0xc80	pmu_clk_counter_7_snapshot_value_31_00	RO	0	32	0 += 32 pmu_clk_counter_7_snapshot_value_31_00
0xc88	pmu_clk_counter_7_counter_value_31_00	RW	0	32	0 += 32 pmu_clk_counter_7_counter_value_31+00

PMU clock counter event mux

The following table describes the PMU clock counter multiplexing.

Table D-6: pmu_clk_counter_#_event_mux

#	Event
0	cycle counter
1	request
2	upload stall
3	l1 allocation
4	l1 update
5	l1 hit
6	l1 miss
7	l1 eviction
8	c1 allocation

#	Event
9	c1 update
10	c1 hit and set
11	c1 hit but not set
12	c1 miss
13	c2 allocation
14	c2 update
15	c2 hit and set
16	c2 hit but not set
17	c2 miss
18	unused
19	unused
20	unused
21	unused
22	unused
23	unused
24	unused
25	unused
26	unused
27	unused
28	unused
29	unused
30	unused
31	unused

PMU clock divider counter event mux

The following table describes the PMU clock divider counter multiplexing.

Table D-7: pmu_clkdiv2_counter_#_event_mux

#	Event
0	cycle counter
1	allocate
2	queue depth
3	waiting for wr data
4	read backlog
5	waiting for mi
6	hazard resolution
7	enqueue
8	arbitrate
9	lrnk turnabout activate
10	prnk turnabout activate
11	read depth

#	Event
12	write depth
13	high high qos depth
14	high qos depth
15	medium qos depth
16	low qos depth
17	activate
18	rdrw
19	refresh
20	training request
21	t mac tracker
22	bk fsm tracker
23	bk open tracker
24	ranks in pwr down
25	ranks in sref
26	client mode stall
27	unused
28	unused
29	unused
30	unused
31	unused

Appendix E Revisions

This appendix describes the technical changes between released issues of this book.

Table E-1: Issue 0000-01

Change	Location
First release.	-

Table E-2: Differences between issue 0000-01 and 0000-02

Change	Location
Added new topic, Client mode and feature_config register, to the DMC-Bing appendix. It describes the bit assignments and the requirement to set bit 8, si_clk_gate_disable, when in Client mode.	DMC-Bing

Table E-3: Differences between issue 0000-02 and 0000-03

Change	Location
Hardware, configuration, register, and appendices content added. Document status has changed from Preliminary FVP-based TRM, to a full silicon-based TRM.	All sections.