Arm® Power Policy Unit

Version 1.1

Architecture Specification

Non-Confidential

ARM DEN 0051E

Arm® Power Policy Unit

Release Information

The following changes have been made to this specification.

Change History

Date	Issue	Confidentiality	Change
15 July 2015	A	Confidential	Beta release.
15 December 2015	B	Confidential	Release 1.0.
26 April 2016	C	Confidential	Release 1.1 - Beta.
4 January 2018	D	Confidential	Release 1.1 – EAC.
23 May 2018	E	Non-Confidential	Non-Confidential Release.

is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

LES-PRE-20349

Contents

Arm® Power Policy Unit

1 Preface 1-1
1.1 About this Specification 1-2
1.2 Using this Specification 1-3
1.3 Conventions 1-4
1.4 Additional Reading 1-5
1.5 Feedback on Documentation 1-6
2 Introduction 2-1
2.1 Power Control Framework 2-2
2.2 Power Policy Unit Overview 2-3
3 PPU Modes 3-1
3.1 PPU Mode Overview 3-2
3.2 PPU Mode Values 3-6
3.3 Power Mode Transitions 3-8
3.4 Power Mode Transition Rules 3-11
3.5 Operating Mode Transitions 3-21
3.6 Operating Mode Transition Rules 3-24
4 Functional Description 4-1
4.1 Interfaces 4-2
4.2 Power Mode Transition Sequences 4-25
4.3 Operating Mode Transition Sequences 4-43
5 Programmers Model 5-1
5.1 Register Summary 5-2
5.2 Register Descriptions 5-3
6 Signal Descriptions 6-1
6.1 Clock and Reset Signals 6-2
6.2 Software Interface Signals 6-3
6.3 Device Interface Signals 6-4
6.4 Device Control Signals 6-5
6.5 PCSM Interface Signals 6-6
7 Configuration Options 7-1
7.1 PPU Configuration 7-2
8 Appendix: AXI LPI Support 8-1
8.1 AXI LPI Support 8-2
9 Appendix: Operating Mode Use Model Examples 9-1
9.1 Ladder Use Model Example – Cache Memory 9-2
9.2 Independent Use Model Example – Multi-threaded Processor Core 9-4
10 Glossary 10-1

11 Revisions 11-1

1 Preface

This preface introduces the Power Policy Unit Architecture Specification. It contains the following sections:

- *About this Specification* on page 1-2.
- *Using this Specification* on page 1-3.
- *Conventions* on page 1-4.
- *Additional Reading* on page 1-5.
- *Feedback on Documentation* on page 1-6.

1.1 About this Specification

This specification describes the *Power Policy Unit* (PPU) architecture. It defines version 1.1 of the PPU architecture.

Version 1.1 of the PPU specification adds operating mode support. This enables component specific support of power modes for functional or higher granularity power control reasons.

Throughout this document, references to *the PPU* or *a PPU* refer to a device that implements this PPU architecture. Unless the context makes it clear that a reference is to an optional or IMPLEMENTATION DEFINED feature of the device, these references describe the requirements of this specification.

1.1.1 Intended Audience

This specification is written for users who want to design, integrate, or program the PPU.

The specification assumes that users are familiar with the Arm Low Power Interfaces. See the *Low Power Interface Specification: Arm® Q-Channel and P-Channel Interfaces* for more information.

1.2 Using this Specification

This specification is organized into the following chapters:

Chapter 1 ***Preface***

Read this for an introduction to this document.

Chapter 2 ***Introduction***

Read this for an introduction to the Power Control Framework and the Power Policy Unit.

Chapter 3 ***PPU Modes***

Read this for information about the power and operating modes of the Power Policy Unit.

Chapter 4 ***Functional Description***

Read this for information about how the Power Policy Unit operates.

Chapter 5 ***Programmers Model***

Read this for a description of the Power Policy Unit register interfaces, and all Power Policy Unit registers.

Chapter 6 ***Signal Descriptions***

Read this for a description of the Power Policy Unit signals.

Chapter 7 ***Configuration Options***

Read this for a description of the Power Policy Unit configuration options.

Chapter 8 ***Appendix: AXI LPI Support***

Read this for information about using the Power Policy Unit with components that have AXI LPI interfaces.

1.3 Conventions

The following describes the conventions used in this document:

Typographical conventions

The following typographical conventions are used:

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

bold Denotes signal names, and is used for emphasis in descriptive lists, where appropriate.

SMALL CAPITALS

Used for a few terms that have specific technical meanings

Timing diagrams

The figure named Key to timing diagram conventions 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.

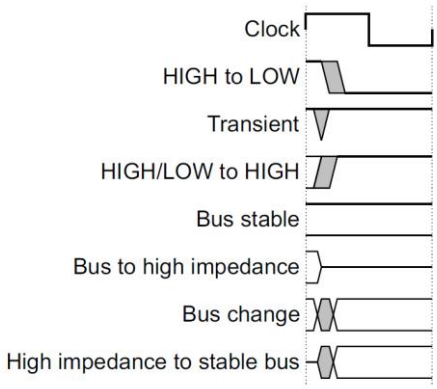

Figure 1-1 Key to timing diagram conventions

Timing diagrams sometimes show single-bit signals as HIGH and LOW at the same time and they look similar to the bus change shown in Key to timing diagram conventions. If a timing diagram shows a single-bit signal in this way then its value does not affect the accompanying description.

Signals

In general this specification does not define hardware signals, but it does include some signal examples and recommendations. 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.

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

Numbers

Numbers are normally written in decimal. Binary numbers are preceded by 0b, and hexadecimal numbers by 0x. To improve readability, long numbers can be written with an underscore separator between every four characters, for example 0xFFFF_0000_0000_0000. Ignore any underscores when interpreting the value of a number.

1.4 Additional Reading

This section lists relevant publications from Arm and third parties.

1.4.1 Arm publications

See the Infocenter, http://infocenter.arm.com for access to the following Arm documentation:

- *Low Power Interface Specification: Arm® Q-Channel and P-Channel Interfaces* (ARM IHI 0068)
- *ARM® AMBA® AXI™ and ACE™ Protocol Specification* (ARM IHI 0022)

For access to the following documentation please contact Arm:

- *ARM® Power Control System Architecture Specification Version 2.0* (ARM DEN 0050C)

1.5 Feedback on Documentation

If you have comments on this documentation, e-mail errata@arm.com. Provide:

- The document title.
- The document number, ARM DEN 0051E.
- The page numbers to which your comments apply.
- A concise explanation of your comments.

Arm also welcomes general suggestions for additions and improvements.

2 Introduction

This section introduces the Power Policy Unit and its features. It contains the following sections:

- *Power Control Framework* on page 2-2
- *Power Policy Unit Overview* on page 2-3

2.1 Power Control Framework

The power control framework is a collection of standard infrastructure components, interfaces, and associated methods that can be used to build the infrastructure necessary for power management of a SoC.

Standard infrastructure components include power and clock control components.

Local interfacing between the infrastructure components and functional components use ARM Q-Channel and P-Channel *Low Power Interfaces* (LPI). Components without support for Arm LPI are managed using an integration layer adaptation approach.

Figure 2-1 shows a high-level illustration of power control framework concepts.

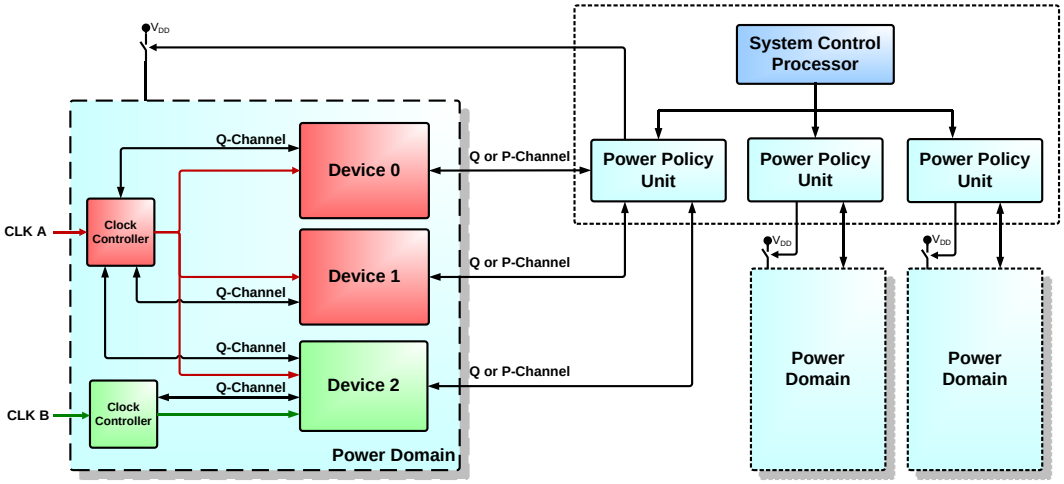

Figure 2-1 Power control framework overview

For power domain control a component known as the *Power Policy Unit* (PPU) is defined. The PPU is fixed function hardware supporting a set of power policies programmed by the System Control Processor (SCP) through a software interface. The PPU interfaces with power domain components, using LPI as needed, to ensure safe power mode transitions.

Clock controllers target components that support high-level clock gating, these include many Arm CoreLink™ system components. This approach enables the clock to be gated at the clock tree root when components are idle.

For a more detailed description of the Power Control Framework, see the *ARM Power Control System Architecture Specification*.

2.2 Power Policy Unit Overview

The PPU is a standard component for abstracting software-controlled power domain policy down to low level hardware control signaling. It enables re-usability by separating device and technology specifics, and the provision of a common software interface.

The PPU has the following interfaces:

- **Software interface:** For high-level policy control and configuration.
- **Device control interface:** For low level device control. It ensures device quiescence and functional control. This includes:
    - The device interface, that consists of one or more LPI.
    - The device controls, that includes clock enables, resets, and isolation control.
- **Power Control State Machine (PCSM) interface:** For controlling low level technology specific power switch and retention controls.

Figure 2-2 illustrates the PPU interfaces.

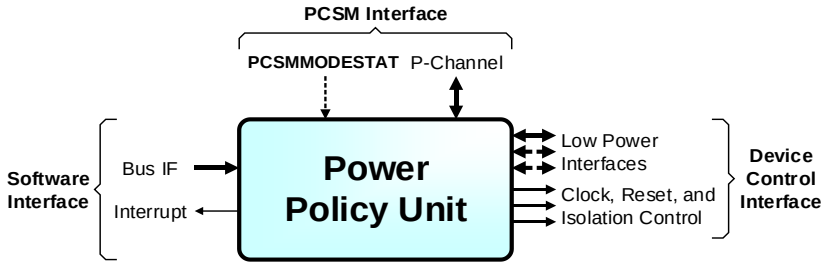

Figure 2-2 PPU Structure

The PPU provides technology-independent hardware and software interfaces for controlling domain power modes in co-ordination with device quiescence. The device interface uses either a single P-Channel or one or more Q-Channels, as described in the *Low Power Interface Specification ARM Q-Channel and P-Channel Interfaces*.

The Power Control State Machine (PCSM) is a technology-dependent state machine for the sequencing of power switch chains and retention controls, that can include RAM and register retention. The PCSM executes power mode changes under PPU direction. The interface between the PPU and the PCSM is a P-Channel.

Figure 2-3 shows a high-level illustration of how the PPU and PCSM controls connect to each other, and to a power gated domain. Dotted lines around components and signal connections indicate these are implementation-dependent.

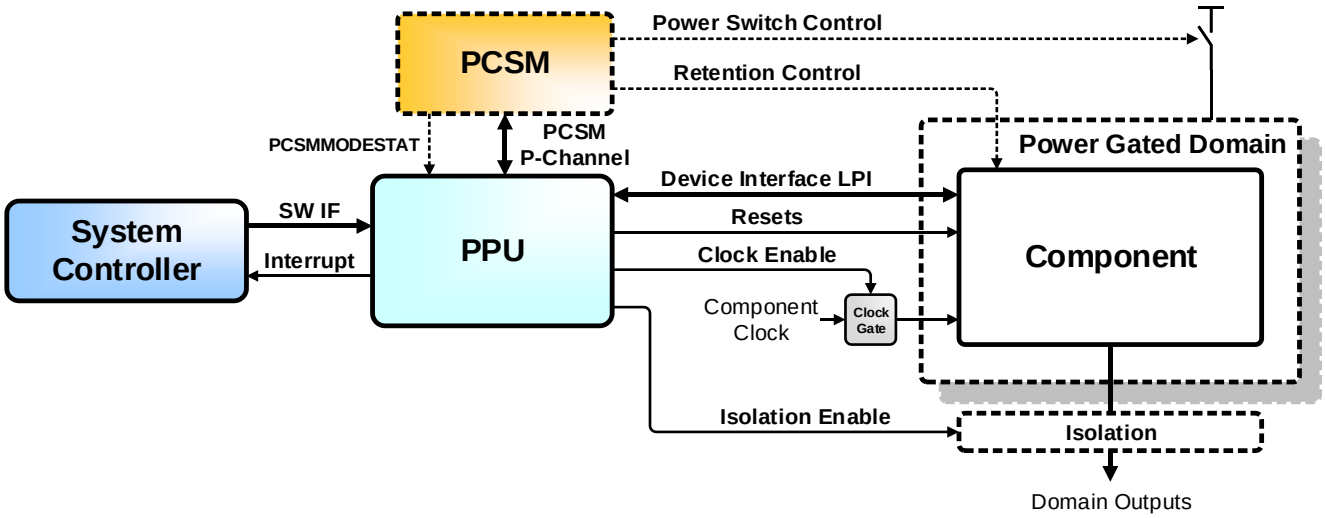

Figure 2-3 Example PPU connections to a power gated domain

2.2.1 PPU Operation

The PPU uses power modes, such as on (ON), off (OFF), and full retention (FULL_RET), to represent the various power conditions of a domain. It has extensive support to reflect the various combinations of logic and memory power states into which a domain can be set.

Software can use these modes as a static policy, a request to enter a mode directly, or a dynamic policy that becomes a minimum mode, so the PPU can autonomously change mode above this minimum based on hardware inputs.

The PPU has the following features, many of which are optional, to make the PPU applicable to multiple power domain scenarios whilst maintaining standard hardware and software interfaces.

2.2.2 PPU v1.1 Additional Features.

PPU v1.1 adds support for operating modes. The operating modes are an optional extension, only supported on a P-Channel PPU.

Operating modes represent configurations of the power modes. The meaning of each operating mode is specific to one or more components within the domain. For further information see *Operating Modes* on page 3-4.

To support operating modes PPU v1.1 also adds interrupts to support detecting the difference between the completion of power mode transitions and the completion of operating mode transitions.

2.2.3 PPU Device Interface Configuration Types

A PPU that uses one or more Q-Channels as the device interface is called a Q-Channel PPU.

A PPU that uses a P-Channel as the device interface is called a P-Channel PPU.

The other interfaces of both types of PPU are the same, including the PCSM P-Channel.

2.2.4 PPU Optional Features

The PPU is a configurable component architecture designed to produce solutions for different power domain scenarios. To this end many of the features are optional or configurable. This section gives an overview of those options.

Software can determine which features the PPU supports by reading the *PPU Identification Register 0 (PPU_IDR0)*, see page 5-22, and the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

Power Mode Support

Most power modes are optional, only those required for the domain are required to be supported. The required and optional power modes are detailed in *Power Modes* on page 3-2.

The default power mode can also be configured, to a limited set of power modes, for further details see *Default Power Policy* on page 7-2.

Operating Mode Support

The number of operating modes can be configured, or they can be removed entirely.

The default operating mode can also be configured, for further details see *Default Operating Policy* on page 7-6.

The dynamic behavior of the PPU operating modes in response to **DEVPACTIVE** inputs can also be configured.

For further information see *Operating Modes* on page 3-4.

Additional Optional Features

The following features are optional, and an implementation can choose to include them or not:

- Power mode entry delay timers: This allows software to configure the delay after the hardware entry conditions are met before a power mode transition takes place.
- Device control delay software configuration: This allows software to configure the time between the device controls, clock enables, isolation and reset.
    - Without this feature delays are still design time configurable.
- Lock: This enables the PPU to be locked when it enters the off or memory retention power modes. This allows a domain to be autonomously switched off but give software control over when it is switched on again. This is used together with the locked interrupt event so software knows when the PPU has been locked in a power mode.
- Retention RAM configuration registers: When these registers exist, they can be used to provide additional information for memory retention power modes.
- Static power mode transition complete interrupt: This additional interrupt added in v1.1 is used to indicate when the programmed static power mode is reached, regardless of the operating mode.
- Static operating mode transition complete interrupt: This additional interrupt added in v1.1 is used to indicate when the programmed static operating mode is reached, regardless of the power mode.
- Support for initialization into MEM_RET when the PPU has been reset/powered-off.

Configuration

The PPU can be further configured with the following parameters:

- P-Channel delay values: Theses are used to configure the timing between PCSM and device interface P-Channel **PSTATE** and **PREQ** signals to aid timing closure, see *P-Channel Delays* on page 7-2.
- Device control delay values: These are used to set the time between the device controls: clock enables, isolation enables, and resets. When device control delay software configuration is supported these values form the default values of these registers. See *Device Control Delay Parameters* on page 7-3.
- Transition control parameters: These configure certain transition control behaviors, and set the default values for the software configuration parameters that set these behaviors, see *Transition Control Parameters* on page 7-3. The behaviors are:
    - If warm reset transitions involve a device interface handshake.
    - If the debug recovery mode, where supported, asserts the power on reset.

3 PPU Modes

This section describes the modes and transitions supported by the PPU. It contains the following sections:

- *PPU Mode Overview* on page 3-2.
- *PPU Mode Values* on page 3-6.
- *Power Mode Transitions* on page 3-8.
- *Power Mode Transition Rules* on page 3-11.
- *Operating Mode Transitions* on page 3-21.
- *Operating Mode Transition Rules* on page 3-24.

3.1 PPU Mode Overview

The PPU supports two mode groups, power modes and operating modes.

Power modes cover the normal combinations of logic and RAM power states for a domain and the associated clock, reset and isolation control.

Operating modes represent configurations of the power modes. The meaning of each operating mode is specific to one or more components within the domain. For further information see *Operating Modes* on page 3-4.

The operating modes are an optional extension, only supported on a P-Channel PPU.

The overall PPU mode is the combination of the power mode and the operating mode.

3.1.1 Power Modes

This section describes the power modes the PPU supports. It must support the on (ON), warm reset (WARM_RST) and off (OFF) modes. Support for other power modes is optional.

The PPU power modes are described in Table 3-1. Whether a domain contains RAM is optional.

Table 3-1 Power modes

Power Mode	Short Name	Logic Mode	RAM Mode	Description
Debug Recovery Reset	DBG_RECOV	ON	ON	Warm reset application with logic and RAM on. This mode is used to enable reset of a component, typically when locked up or non-functional, whilst retaining some or all component state through the reset for later debug analysis.
Warm Reset	WARM_RST	ON	ON	Warm reset application with logic and RAM on.
On	ON	ON	ON	Logic on with RAM on, component is functional.
Functional Retention	FUNC_RET	ON	RET	Logic on with RAM retained, component is functional.
Memory Off	MEM_OFF	ON	OFF	Logic on with RAM off, component is functional.
Full Retention	FULL_RET	RET	RET	Logic and RAM in retention.
Logic Retention	LOGIC_RET	RET	OFF	Logic retention with RAM off.
Emulated Memory Retention	MEM_RET_EMU	ON	ON	Logic on with RAM on. This mode is used to emulate the functional condition of MEM_RET without removing power.
Memory Retention	MEM_RET	OFF	RET	Logic off with RAM retained.
Emulated Off	OFF_EMU	ON	ON	Logic on with RAM on. This mode is used to emulate the functional condition of OFF without removing power.
Off	OFF	OFF	OFF	Logic off and RAM off.

In MEM_RET, FUNC_RET, and FULL_RET power modes the configuration of the memory retention can be specified by software programming if the retention RAM configuration registers are supported, see *Functional Retention RAM Configuration Register (PPU_FUNRR)* on page 5-18, *Full Retention RAM Configuration Register (PPU_FULRR)* on page 5-18, and *Memory Retention RAM Configuration Register (PPU_MEMRR)* on page 5-19.

Off (OFF)

In this power mode logic and RAM are off and all state is lost.

Emulated Off (OFF_EMU)

In this power mode, the functional conditions of the OFF power mode are replicated, but the power to the logic and RAM remains on.

The reset, and other device control, application is different from OFF to allow some portions of the domain to be functional. For example, processor debug state and access can be maintained while emulating the state loss, through reset of functional logic, of OFF.

For further information about controlling this power mode see *Power Mode Emulation Programming* on page 4-4.

Memory Retention (MEM_RET)

In this power mode logic is off. RAM state is retained.

The scope of the retention in the domain is IMPLEMENTATION DEFINED. Retention can be either full or partial. Software configures the extent of the memory retention using the *Memory Retention RAM Configuration Register (PPU_MEMRR)*, see page 5-19.

Emulated Memory Retention (MEM_RET_EMU)

In this power mode the functional conditions of MEM_RET are replicated, but the power to the logic and RAM remains on.

The reset, and other device control, application is different from MEM_RET to allow some portions of the domain to be functional. This allows, for instance, processor debug state and access to be maintained while emulating the state loss, through reset of functional logic, of MEM_RET.

For further information about controlling this power mode see *Power Mode Emulation Programming* on page 4-4.

Logic Retention (LOGIC_RET)

In this power mode logic is non-functional, but state is retained. RAMs are off and RAM state is lost.

Note

This mode can only be supported by a P-Channel PPU.

Full Retention (FULL_RET)

In this power mode logic is non-functional, but state is retained. RAM state can be retained.

The scope of the retention in the domain is IMPLEMENTATION DEFINED. Retention can be either full or partial. Software configures the extent of the memory retention using the *Full Retention RAM Configuration Register (PPU_FULRR)*, see page 5-18.

Memory Off (MEM_OFF)

In this power mode logic is on and functional. RAMs are off and RAM state is lost.

Functional Retention (FUNC_RET)

In this power mode logic is on and functional. RAM is non-functional, but state is retained.

The scope of the retention in the domain is IMPLEMENTATION DEFINED. Retention can be either full or partial. Software configures the extent of the memory retention using the *Functional Retention RAM Configuration Register (PPU_FUNRR)*, see page 5-18

On (ON)

In this power mode logic and RAMs are on and functional.

Warm Reset (WARM_RST)

In this power mode logic and RAMs are on but specific resets are asserted.

For more information on the resets that are applied in this mode see *Device Resets* on page 4-18.

Debug Recovery Reset (DBG_RECOV)

In this power mode logic and RAMs are on but specific resets are asserted.

For more information on the resets that are applied in this power mode see *Device Resets* on page 4-18.

This power mode is used to reset a domain in the event of a component lock-up or unexpected behaviour. This mode allows specified component state, such as memory contents, to be retained through a reset cycle for debugging purposes.

Note

This mode can only be supported by a P-Channel PPU.

3.1.2 Operating Modes

Operating modes represent configurations of the power modes. The meaning of each operating mode is specific to one or more components within the domain.

Some examples of uses for operating modes are:

- To enable multiple RAM configurations:
    - For example, resizing caches while a component is active, to save leakage power by powering off some RAM instances.
- Thread management within multi-thread processor cores.
    - Ensures correct thread management as interrupts for a logically powered off thread are only available to the power control infrastructure outside of the processor.
- Configuration and access control management, for example, to enable save/restore operations.

The PPU can change operating modes through direct programming and, when dynamic transitions are enabled, in response to **PACTIVE** changes based on defined use models. For more details on the supported use models, see *Operating Mode Transition Rules* on page 3-24.

Note

In some scenarios, such as thread management, operating modes represent a configuration of the component which does not require any change in power state. For these cases, a transition of the operating mode alone does not require a PCSM P-Channel handshake.

However, in the case of operating modes representing RAM configurations, a PCSM P-Channel handshake is required to configure the RAM power state.

Transitions between operating modes occur when in the ON power mode. However, the operating modes can maintain context in other power modes. For example, a RAM operating mode still has context in a retention power mode such as FULL_RET.

However, in the OFF, OFF_EMU, WARM_RST, and DBG_RECOV power modes the operating modes do not have any context.

Figure 3-1 shows an example combination of all power modes and three RAM operating modes. It shows the power modes where the operating modes have context. It does not show the allowed transitions between them.

Although the MEM_RET_EMU power mode turns on all RAMs regardless of the operating mode, it still has operating mode context as it can provide functional control, for example, invalidation of RAM when entering and exiting the power mode.

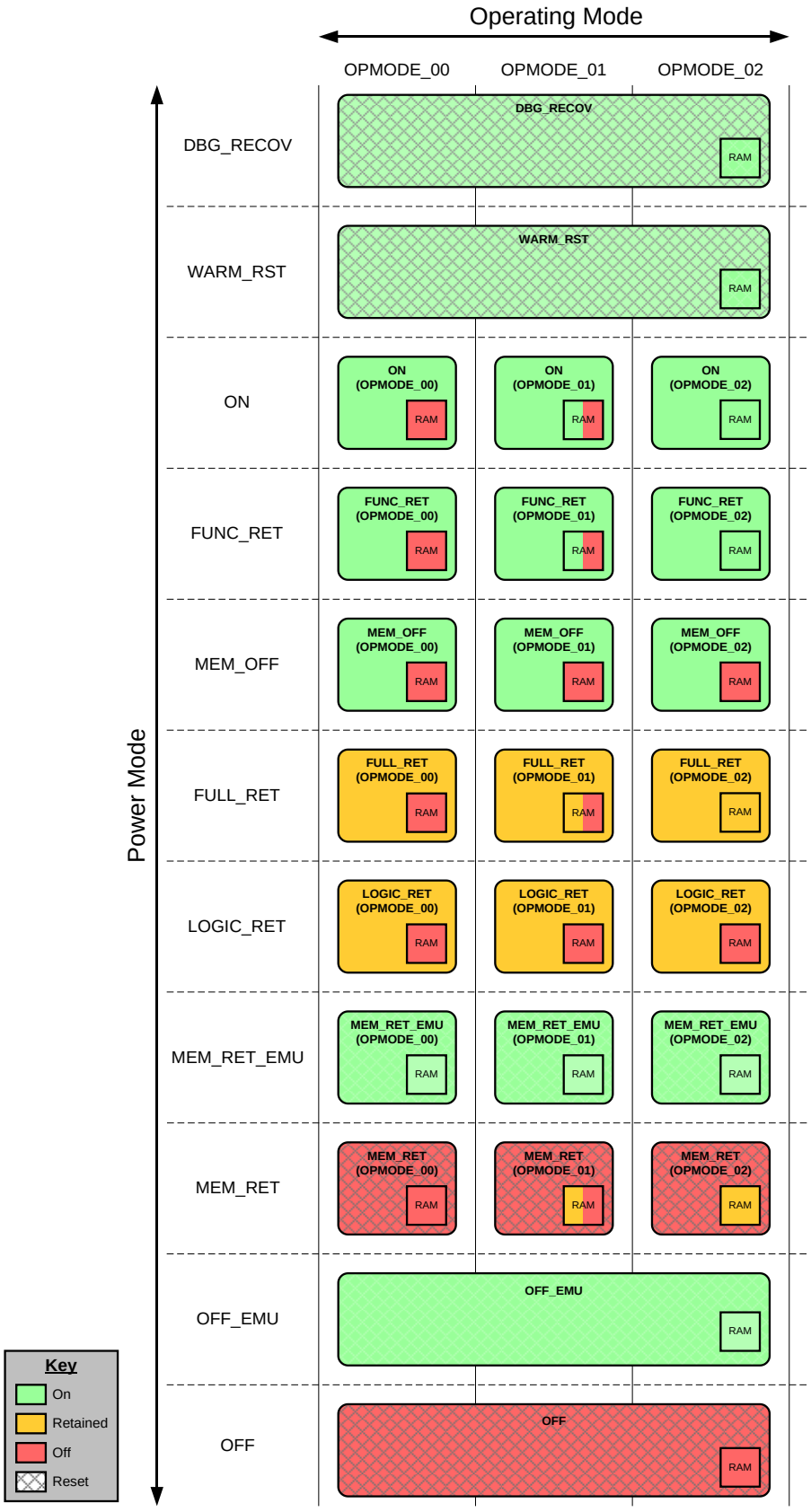

Figure 3-1 – Power and operating mode matrix for three operating modes

3.2 PPU Mode Values

3.2.1 Power Mode Values

Table 3-2 shows how the PPU power modes relate to PPU parameters.

The parameters shown in Table 3-2 are described in the following sections:

- **DEVPACTIVE** and **DEVPSTATE** are described in *Device P-Channel Interface* on page 4-11.
- The PPU_PWPR.PWR_POLICY register field is described in *Power Policy Register (PPU_PWPR)* on page 5-3.
- **PCSMPSTATE** is described in *PPU to PCSM Interface* on page 4-22.
- **PPUHWSTAT** is described in *PPUHWSTAT* on page 4-15.

Table 3-2 Power mode enumeration

Power Mode	DEVPACTIVE bit	DEVPSTATE [3:0] / PPU_PWPR.PWR_POLICY	PCSMPSTATE [3:0]	PPUHWSTAT [15:0] [b]	Priority
DBG_RECOV	10	0b1010	0b1000 [a]	0b0000_0100_0000_0000	High
WARM_RST	9	0b1001	0b1000 [a]	0b0000_0010_0000_0000	|
ON	8	0b1000	0b1000	0b0000_0001_0000_0000	|
FUNC_RET	7	0b0111	0b0111	0b0000_0000_1000_0000	|
MEM_OFF	6	0b0110	0b0110	0b0000_0000_0100_0000	|
FULL_RET	5	0b0101	0b0101	0b0000_0000_0010_0000	|
LOGIC_RET	4	0b0100	0b0100	0b0000_0000_0001_0000	|
MEM_RET_EMU	3	0b0011	0b1000 [a]	0b0000_0000_0000_1000	|
MEM_RET	2	0b0010	0b0010	0b0000_0000_0000_0100	|
OFF_EMU	1	0b0001	0b1000[a]	0b0000_0000_0000_0010	|
OFF	0	0b0000	0b0000	0b0000_0000_0000_0001	Low

[a] In these power modes the PPU instructs the PCSM to have the same conditions as ON.

[b] **PPUHWSTAT[15:11]** are reserved and are always 0b00000

3.2.2 Operating Mode Values

Table 3-3 shows the operating mode **DEVPSTATE**, **PCSMPSTATE**, and **PPUHWSTAT** values supported by the PPU.

Table 3-3 Operating mode enumeration

Operating Mode	Short Name	DEVPSTATE [7:4] / PCSMPSTATE [7:4] / PPU_PWPR.OP_POLICY	PPUHWSTAT [31:16]	Priority
Operating Mode 15	OPMODE_15	0b1111	0b1000_0000_0000_0000	High
Operating Mode 14	OPMODE_14	0b1110	0b0100_0000_0000_0000	|
Operating Mode 13	OPMODE_13	0b1101	0b0010_0000_0000_0000	|
Operating Mode 12	OPMODE_12	0b1100	0b0001_0000_0000_0000	|
Operating Mode 11	OPMODE_11	0b1011	0b0000_1000_0000_0000	|
Operating Mode 10	OPMODE_10	0b1010	0b0000_0100_0000_0000	|
Operating Mode 9	OPMODE_09	0b1001	0b0000_0010_0000_0000	|
Operating Mode 8	OPMODE_08	0b1000	0b0000_0001_0000_0000	|
Operating Mode 7	OPMODE_07	0b0111	0b0000_0000_1000_0000	|
Operating Mode 6	OPMODE_06	0b0110	0b0000_0000_0100_0000	|
Operating Mode 5	OPMODE_05	0b0101	0b0000_0000_0010_0000	|
Operating Mode 4	OPMODE_04	0b0100	0b0000_0000_0001_0000	|
Operating Mode 3	OPMODE_03	0b0011	0b0000_0000_0000_1000	|
Operating Mode 2	OPMODE_02	0b0010	0b0000_0000_0000_0100	|
Operating Mode 1	OPMODE_01	0b0001	0b0000_0000_0000_0010	|
Operating Mode 0	OPMODE_00	0b0000	0b0000_0000_0000_0001	Low

Table 3-3 does not show **DEVPACTIVE** bit associations to operating modes as these depend on the **DEVPACTIVE** use model supported. For further information see *Operating Mode Transition Rules* on page 3-24.

If the OPMODE_PCSM_SPT_CFG parameter is 0 then PCSM transitions for operating mode only transitions are disabled and **PCSMPSTATE[7:4]** is tied to OPMODE_00. For further information see *Operating Mode PCSM Transition Configuration* on page 7-7.

The OFF, OFF_EMU, WARM_RST and DBG_RECOV power modes do not have operating mode context. Therefore, for these modes:

- Device P-Channel transitions to these power modes set the operating mode, on **DEVPSTATE[7:4]**, to OPMODE_00.
- PCSM P-Channel transitions to these power modes set the operating mode, on **PCSMPSTATE[7:4]**, to the highest operating mode the PPU supports.

MEM_RET_EMU has operating mode context, so the operating mode is reflected on **DEVPSTATE[7:4]**. However, as MEM_RET_EMU is emulating memory retention all parts of the domain must be on, so **PCSMPSTATE[7:4]** is set to the highest operating mode supported by the PPU.

3.3 Power Mode Transitions

This section describes the allowed transitions between power modes.

3.3.1 Static and Dynamic Transitions

Power mode transitions are programmed to be either static or dynamic.

For static transitions software sets the policy as the required power mode. Hardware signaling is used to indicate when components within the power domain are ready to enter the required power mode, and therefore when the PPU should attempt the power mode transition.

For dynamic transitions software sets the policy as a minimum power mode. Hardware signaling is used by the PPU to indicate when it should attempt transitions to power modes at or above this minimum.

Rules for these different types of transitions are described in *Power Mode Transition Rules* on page 3-11.

3.3.2 Q-Channel PPU Power Mode Transitions

Figure 3-2 shows the allowed power mode transitions for a Q-Channel PPU.

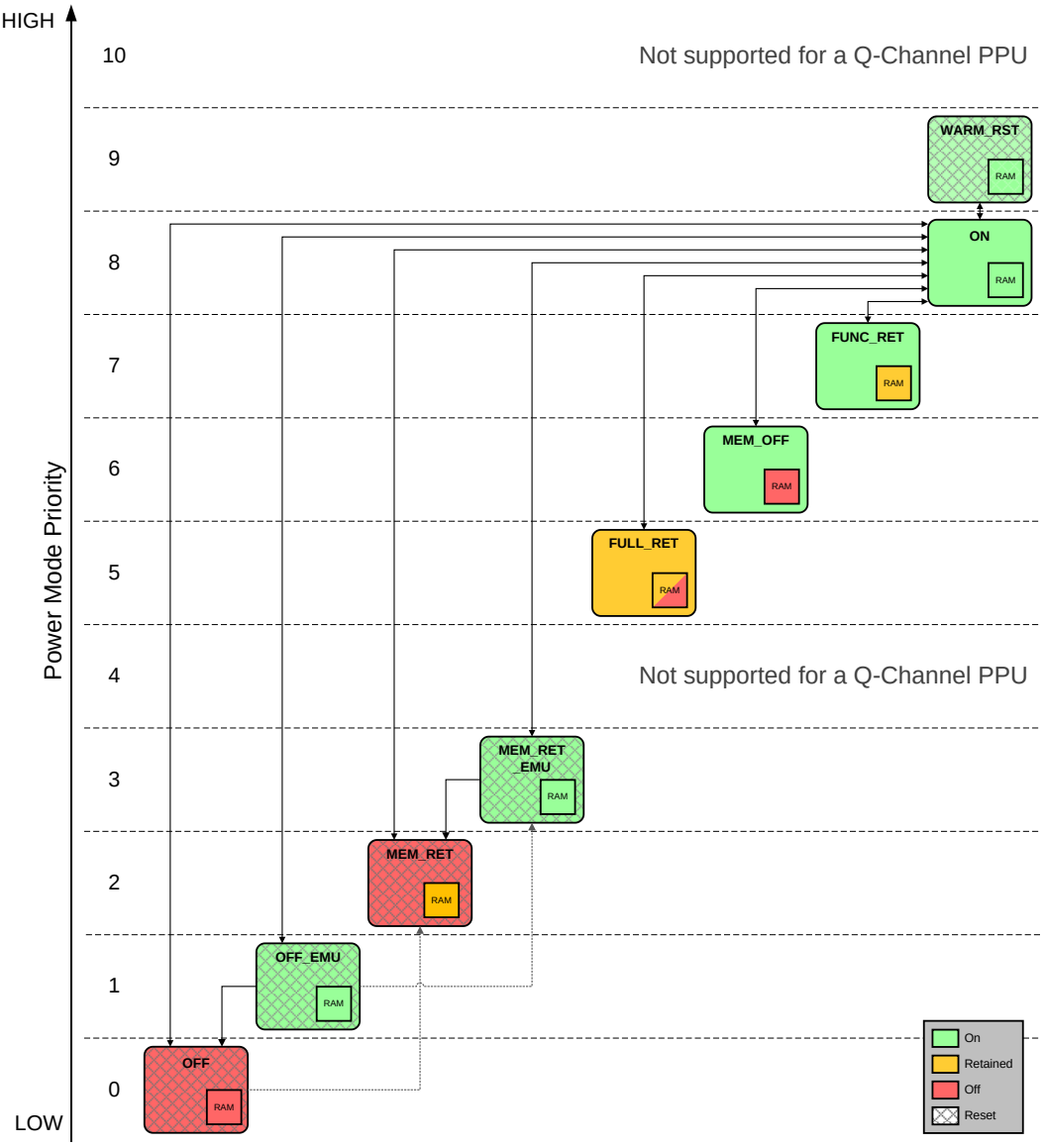

Figure 3-2 Q-Channel PPU power mode transitions

Although multiple power modes are supported with a Q-Channel PPU it only transitions between ON and the power mode programmed in the PPU_PWPR register.

The only exceptions to this are transitions which do not perform a device interface Q-Channel handshake, these transitions are:

- From OFF_EMU to OFF.
- From MEM_RET_EMU to MEM_RET.
- From OFF to MEM_RET.
- From OFF_EMU to MEM_RET_EMU.

For these transitions the component is not aware the transition is taking place.

For any transitions from either OFF or OFF_EMU, to either MEM_RET or MEM_RET_EMU the transition is made without going through ON.

Whether direct transitions from OFF to MEM_RET and from OFF_EMU to MEM_RET_EMU are supported can be configured, see *OFF to MEM_RET Direct Transition Configuration* on page 7-5.

Changes in the programmed power mode policy of the PPU, where there are consequences on component function, must be agreed between the controller and the component using a method other than the device interface Q-Channel, as specific power mode information cannot be communicated on the Q-Channel.

For example, before changing the programmed power mode policy from FULL_RET to OFF, an agreement might be needed between the component and the element controlling the PPU to ensure the correct steps are taken before entering OFF.

3.3.3 P-Channel PPU Power Mode Transitions

Figure 3-3 shows the allowed power mode transitions for a P-Channel PPU.

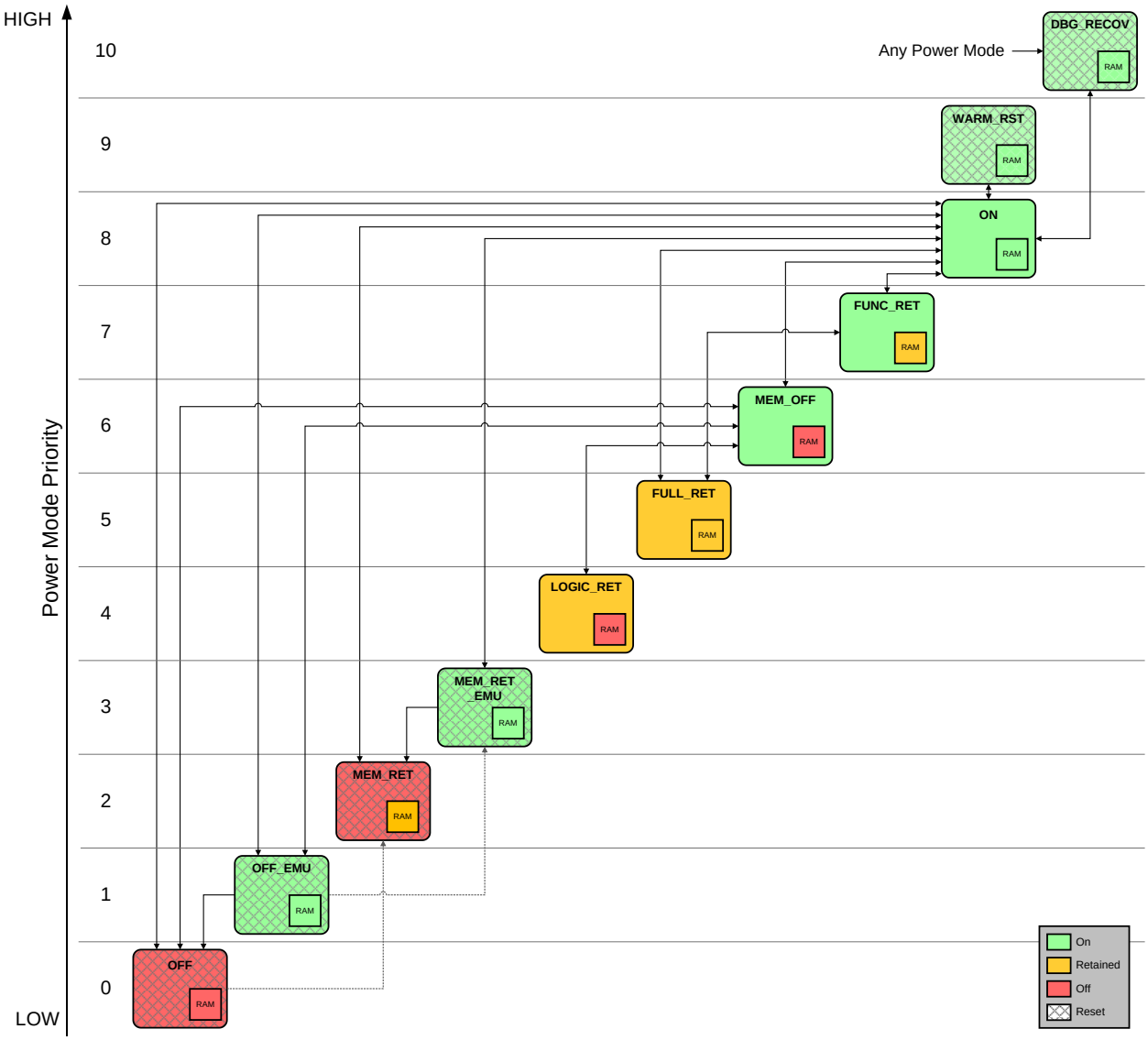

Figure 3-3 P-Channel PPU power mode transitions

For some power mode transitions, there is no device interface P-Channel transition, so the component is not aware the change is taking place. These transitions are listed below:

- From OFF to MEM_RET.
- From OFF_EMU to MEM_RET_EMU.

Whether direct transitions from OFF to MEM_RET and from OFF_EMU to MEM_RET_EMU are supported can be configured, see *OFF to MEM_RET Direct Transition Configuration* on page 7-5.

3.4 Power Mode Transition Rules

This section describes how the PPU initiates power mode transitions depending on the device interface signalling and software configuration.

3.4.1 Power Mode DEVACTIVE Inputs

The power mode **DEVACTIVE** inputs are used to initiate and control power mode transitions in combination with software settings. This section describes how these inputs can be modified by software control.

Power Mode DEVACTIVE Input Enables

The effect of the power mode **DEVACTIVE** inputs on transitions is controlled using PPU.PWCR.PWR_DEVACTIVEEN.

Disabled power mode **DEVACTIVE** inputs are always assumed to be LOW for the purposes of transition control.

For further information on these controls see *DEVACTIVE Enables* on page 4-13.

Power Mode DEVACTIVE Inputs and Emulation Enable

The emulation enable software control, PPU_PMER.EMU_EN, allows software to request entry into the OFF_EMU and MEM_RET_EMU power modes instead of OFF and MEM_RET respectively.

Modifying the emulation enable also influences the **DEVACTIVE** inputs dependent on the type of PPU. For further information see *Emulation Enable Effect on Power Mode DEVACTIVE Inputs in a* Q-Channel PPU on page 3-11, and *Emulation Enable Effect on Power Mode DEVACTIVE Inputs in a* P-Channel PPU on page 3-11 for more details.

If neither OFF_EMU nor MEM_RET_EMU is supported PPU_PMER is reserved.

Emulation Enable Effect on Power Mode DEVACTIVE Inputs in a Q-Channel PPU

The emulation enable has no effect on **DEVQACTIVE** inputs for a Q-Channel PPU.

Emulation Enable Effect on Power Mode DEVACTIVE Inputs in a P-Channel PPU

For a P-Channel PPU, when PPU_PMER.EMU_EN is set to 0b1 the internal view of the **DEVPACTIVE** inputs for the OFF_EMU and MEM_RET_EMU power modes is modified.

This internal **DEVPACTIVE** view is used to determine which transitions are made.

When PPU_PMER.EMU_EN is set to 0b1:

- The internal OFF_EMU **DEVPACTIVE** is set HIGH.
- The internal MEM_RET_EMU **DEVPACTIVE** becomes a logical OR of the MEM_RET and MEM_RET_EMU **DEVPACTIVE** inputs

Table 3-4 shows the effect of the EMU_EN bit on the OFF_EMU **DEVPACTIVE**.

Table 3-4 Effect of EMU_EN on the OFF_EMU DEVPACTIVE

PPU_PMER.EMU_EN	Input OFF_EMUPACTIVE	Internal OFF_EMU PACTIVE
0	0	0
0	1	1
1	X	1

Table 3-5 shows the effect of the EMU_EN bit on the MEM_RET_EMU **DEVPACTIVE**.

Table 3-5 Effect of EMU_EN and DEVPACTIVE inputs on the MEM_RET_EMU DEVPACTIVE

PPU_PMER. EMU_EN	Input MEM_RET PACTIVE	Input MEM_RET_EMU PACTIVE	Internal MEM_RET_EMU PACTIVE
0	X	0	0
0	X	1	1
1	0	0	0
1	1	X	1
1	X	1	1

3.4.2 Static Power Mode Transitions

Transitions to lower priority power modes are delayed until the relevant power mode **DEVACTIVE** inputs are LOW. This is true even if the PPU must go through an intermediate power mode to reach the programmed policy.

For example, a component in FULL_RET cannot make a direct transition to OFF, it must pass through ON. However, all power mode **DEVACTIVE** inputs between FULL_RET and OFF must be LOW before the PPU begins the intermediate transition to ON.

The power mode **DEVACTIVE** inputs do not delay transitions to higher priority power modes.

Note

There are some exceptions to these rules for a Q-Channel PPU transitioning between ON and WARM_RST, for more information see *Static Power Mode Transitions with a Q-Channel PPU* on page 3-12.

For further information on:

- Allowed transitions, see *Power Mode Transitions* on page 3-8.
- Intermediate transitions. see *Intermediate Power Mode Transitions* on page 3-19.
- Relevant **DEVACTIVE** inputs, see *Static Power Mode Transitions with a* Q-Channel PPU on page 3-12 and *Static Power Mode Transitions with a P-Channel PPU* on page 3-13.

Static Power Mode Transitions with a Q-Channel PPU

For transitions between ON and WARM_RST there is no dependence on the **DEVQACTIVE** inputs.

For transitions from ON, other than to WARM_RST, all **DEVQACTIVE** inputs must be LOW to begin a transition.

Effect of Emulation Enable on Q-Channel PPU Power Mode Transitions

When the SW emulation enable bit (PPU_PMER.EMU_EN) is set to 0b1:

- A transition that normally goes to OFF instead goes to OFF_EMU.
- A transition that normally goes to MEM_RET instead goes to MEM_RET_EMU.

The PPU transitions from OFF_EMU to OFF when all the following conditions are met:

- PPU_PWPR.PWR_POLICY is set to OFF.
- All **DEVQACTIVE** inputs are LOW.
- EMU_EN is set to 0b0.

The PPU transitions from MEM_RET_EMU to MEM_RET when all the following conditions are met:

- PPU_PWPR.PWR_POLICY is set to MEM_RET.
- All **DEVQACTIVE** inputs are LOW.
- EMU_EN is set to 0b0.

In both the above cases the power mode entry delay timers have no effect, for more information on these see *Power Mode Entry* Delay Timers on page 4-14.

Static Power Mode Transitions with a P-Channel PPU

For transitions to lower-priority power modes, all relevant **DEVPACTIVE** inputs above the programmed power mode policy, set in PPU_PWPR.PWR_POLICY, must be LOW. There are some exceptions to this rule for the MEM_RET_EMU and OFF_EMU **DEVPACTIVE** inputs. For more information see *Effect of Emulation DEVPACTIVEs on P-Channel PPU Static Power Mode Transitions* on page 3-13.

Transitions to higher-priority power modes are not dependent on the **DEVPACTIVE** inputs.

The relevant **DEVPACTIVE** inputs are defined by the following rules:

- **DEVPACTIVE** inputs for power modes with higher priority than the current power mode:
    - Only those for power modes with static support are considered.
- **DEVPACTIVE** inputs for power modes with equal or lower priority than the current power mode, but higher priority than the PPU_PWPR.PWR_POLICY setting:
    - All are considered.
- **DEVPACTIVE** inputs for power modes with priority equal to or lower than the PPU_PWPR.PWR_POLICY setting:
    - All are ignored.

Figure 3-4 illustrates the **DEVPACTIVE** signals that are considered.

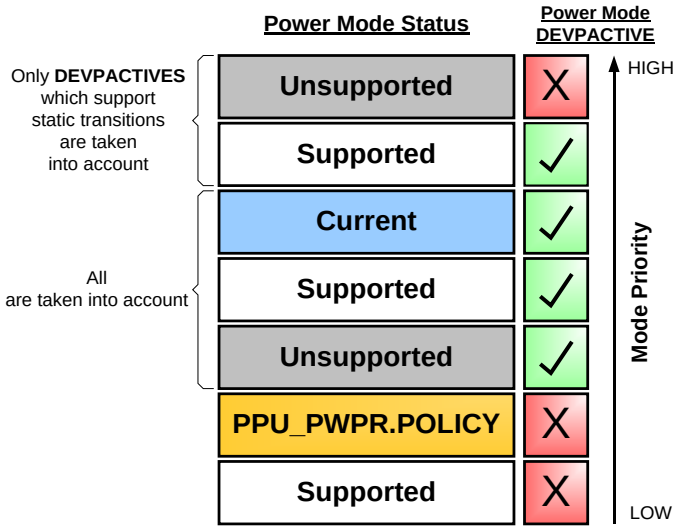

Figure 3-4 Example of relevant static mode DEVPACTIVE inputs

Effect of Emulation DEVPACTIVEs on P-Channel PPU Static Power Mode Transitions

The PPU transitions to OFF_EMU rather than OFF when:

- PPU_PWPR.PWR_POLICY is set to OFF.
- The OFF_EMU **DEVPACTIVE** is HIGH.
- All other relevant higher priority **DEVPACTIVE** inputs are LOW.

The PPU transitions from OFF_EMU to OFF when all the following conditions are met:

- PPU_PWPR.PWR_POLICY is set to OFF.
- The OFF_EMU **DEVPACTIVE** is LOW.
- All relevant higher priority **DEVPACTIVE** inputs are LOW.

The PPU transitions to MEM_RET_EMU rather than MEM_RET when all the following conditions are met:

- PPU_PWPR.PWR_POLICY is set to MEM_RET.
- The MEM_RET_EMU **DEVPACTIVE** is HIGH.
- All other relevant higher priority **DEVPACTIVE** inputs are LOW.
    - For transitions from higher power modes

The exception to the above rule is if the PPU is in OFF, with OFF to MEM_RET direct transitions enabled, then it will transition instead to MEM_RET.

The PPU transitions from MEM_RET_EMU to MEM_RET when all the following conditions are met:

- PPU_PWPR.PWR_POLICY is set to MEM_RET.
- The MEM_RET_EMU **DEVPACTIVE** is LOW.
- All relevant higher priority **DEVPACTIVE** inputs are LOW.

For the transitions from OFF_EMU to OFF, and MEM_RET_EMU to MEM_RET, the power mode entry delay timers have no effect, for more information on these see *Power Mode Entry Delay Timers* on page 4-14.

Note

The OFF_EMU and MEM_RET **DEVPACTIVE** inputs are modified by the PPU_PWPR.EMU_EN setting, see *Emulation Enable Effect on Power Mode DEVACTIVE Inputs in a P-Channel PPU* on page 3-11.

3.4.3 Dynamic Power Mode Transitions

Dynamic power mode transitions can be initiated by changes on the power mode **DEVACTIVE** inputs in addition to changes to the programmed policy.

Note

Support for dynamic power mode transitions is optional. For example, a domain might support dynamic and static transitions to FUNC_RET but only static transitions to OFF. Supported power modes can be determined by software, see the *PPU Identification Register 0 (PPU_IDR0)* on page 5-22.

When power mode dynamic transitions are enabled or disabled, the PPU completes any ongoing transition before the new setting takes effect. The PPU_PWSR.PWR_DYN_STATUS bit is updated when the new setting takes effect.

Lock

This feature allows a domain to dynamically go to OFF, or into MEM_RET, whilst allowing software to maintain control over when it can transition to a higher power mode.

With the lock feature and power mode dynamic transitions enabled, when the PPU reaches a lockable power mode it remains in that power mode until it is unlocked. This is the case even if the **DEVACTIVE** inputs, or software set minimum policy, are requesting a higher priority power mode.

The lockable power modes are:

- OFF.
- OFF_EMU.
- MEM_RET.
- MEM_RET_EMU.

When locked the PPU can transition:

- From OFF_EMU to OFF.
- From MEM_RET_EMU to MEM_RET.

This feature is controlled using the PPU_PWPR.LOCK_EN bit. For more details see the *Power Policy Register (PPU_PWPR)* on page 5-3.

Whether the PPU is currently locked can be read in the PPU_PWSR.LOCK_STATUS bit. For more details see the *Power Status Register (PPU_PWSR)* on page 5-5.

Note

Support for lock functionality is optional. Whether it is supported can be determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

Locking the PPU

The PPU is locked when both of the following conditions are met:

- The lock feature is enabled (PPU_PWPR.LOCK_EN is set to 0b1).
- Power mode dynamic transitions are enabled (PPU_PWPR.PWR_DYN_EN is set to 0b1).

And any of the following conditions occur:

- The PPU transitions to a lockable power mode, except when transitioning:
    - From OFF_EMU to OFF.
    - From MEM_RET_EMU to MEM_RET.
- The PPU_PWPR is written with LOCK_EN and PWR_DYN_EN set to 0b1 and:
    - The PPU is in a lockable power mode.
    - The PPU is transitioning from:
        - OFF_EMU to OFF.
        - MEM_RET_EMU to MEM_RET.

  LOCK_EN and PWR_DYN_EN do not need to have been 0b1 before the write.

  The PPU will be locked even if it has not performed any transitions since the last unlock.
- A transition from one of the lockable power modes is denied, except when transitioning:
    - From OFF_EMU to OFF.
    - From MEM_RET_EMU to MEM_RET.

Unlocking the PPU

The PPU is unlocked when any of the following conditions occur:

- Software writes a one to the PPU_UNLK.UNLOCK bit.
- Either the PPU_PWPR.LOCK_EN, or the PPU_PWPR.PWR_DYN_EN, is set to 0b0.

Writing the UNLOCK bit unlocks the PPU with a single write without disabling the feature, so the PPU locks again when it re-enters a lockable power mode.

The PPU locks again when it meets the conditions listed in *Locking the PPU* on page 3-15.

Dynamic Power Mode Transitions with a Q-Channel PPU

When power mode dynamic transitions are enabled, a Q-Channel PPU can move between ON and the programmed power mode policy in PPU_PWPR. It performs these transitions as requested by the **DEVQACTIVE** inputs.

When in ON, all **DEVQACTIVE** signals set LOW starts a transition to the programmed power mode policy. When in the programmed power mode policy, a HIGH level on any **DEVQACTIVE** input starts a transition to ON.

The exception to this rule is WARM_RST. Dynamic transitions to and from this power mode are not supported for a Q-Channel PPU.

Effect of Emulation Enable on Q-Channel PPU Power Mode Transitions

When the SW emulation enable bit (PPU_PMER.EMU_EN) is set to 0b1:

- A transition which normally goes to OFF instead goes to OFF_EMU.
- A transition which normally goes to MEM_RET instead goes to MEM_RET_EMU.

The PPU transitions from OFF_EMU to OFF when all the following conditions are met:

- PPU_PMER.EMU_EN is set to 0b0.
- PPU_PWPR.PWR_POLICY is set to OFF.
- All **DEVQACTIVE** inputs are LOW.

The PPU transitions from MEM_RET_EMU to MEM_RET when all the following conditions are met:

- PPU_PMER.EMU_EN is to 0b0.
- PPU_PWPR.PWR_POLICY is set to MEM_RET.
- All **DEVQACTIVE** inputs are LOW.

In both the above cases the power mode entry delay timers have no effect, for more information see *Power Mode Entry* Delay Timers on page 4-14.

Note

For a Q-Channel PPU in OFF or MEM_RET, with dynamic transitions enabled, setting PPU_PMER.EMU_EN alone does not cause a transition to happen.

Dynamic Power Mode Transitions with a P-Channel PPU

This section describes the transition behaviour when power mode dynamic transitions are enabled, PPU_PWPR.PWR_DYN_EN is set to 0b1.

When power mode dynamic transitions are enabled, a P-Channel PPU can move between power modes of equal or higher priority than the programmed power mode policy.

The exception to this rule is DBG_RECOV, dynamic transitions to this power mode are not supported.

It performs these transitions as requested by the power mode **DEVPACTIVE** inputs. The programmed power mode policy, PPU_PWPR.PWR_POLICY, is the lowest priority power mode allowed.

The target power mode is the highest priority of:

- The power mode programmed in PPU_PWPR.PWR_POLICY.
- The power mode requested by the **DEVPACTIVE** inputs.

The **DEVPACTIVE** inputs used for this calculation are defined by the following rules:

- **DEVPACTIVE** inputs for power modes with higher priority than the current power mode:
    - Only those for power modes with dynamic support are considered.
- **DEVPACTIVE** inputs for power modes with equal or lower priority than the current power mode, but higher priority than the PPU_PWPR.PWR_POLICY setting:
    - All are considered.
- **DEVPACTIVE** inputs for power modes with priority equal to or lower than the PPU_PWPR.PWR_POLICY setting:
    - All are ignored.

Figure 3-5 illustrates the **DEVPACTIVE** signals that are considered.

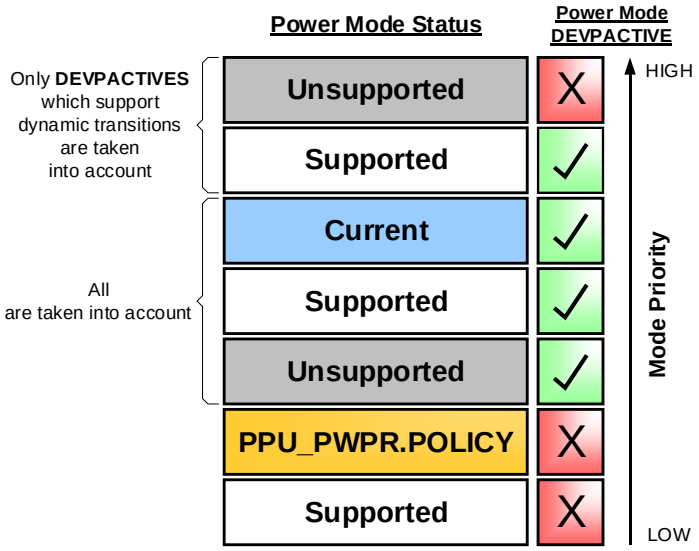

Figure 3-5 Relevant DEVPACTIVE inputs with dynamic transitions enabled

If the target power mode is not supported for dynamic mode transitions then the target mode becomes the next highest priority mode, up to ON, that is supported for dynamic transitions.

If there is no valid supported power mode above the target which supports dynamic transitions, then no transition takes place.

The following are examples of the behavior. Power mode dynamic transitions are enabled in all these scenarios.

Example 1:

- The PPU is in OFF.
- PPU_PWPR.PWR_POLICY is programmed to OFF.
- MEM_OFF is not supported for dynamic transitions.
- The MEM_OFF **DEVPACTIVE** input is HIGH.
- All other **DEVPACTIVE** inputs are LOW,
    - The PPU remains in OFF.

Example 2:

- The PPU is in FUNC_RET.
- PPU_PWPR.PWR_POLICY is programmed to FUNC_RET.
- WARM_RST is not supported for dynamic transitions.
- The ON and WARM_RST **DEVPACTIVE** inputs are HIGH
    - The PPU moves to ON.

Example 3:

- The PPU is in ON.
- PPU_PWPR.PWR_POLICY is programmed to OFF.
- MEM_OFF is not supported for dynamic transitions.
- All **DEVPACTIVE** inputs for power modes with a higher priority than OFF, including MEM_OFF, are LOW.
    - The PPU moves to OFF.

Intermediate Power Mode Transitions

A direct transition to the requested power mode might not be possible. In this case transitions to intermediate power modes are allowed provided they meet the following conditions:

- When the power mode requested is lower priority than the current power mode, the PPU can move to an intermediate power mode with higher priority than the requested power mode.
- When the power mode requested is higher priority than the current power mode, the PPU can move to an intermediate power mode with higher priority than the current power mode.

Note

To allow a dynamic transition to a power mode that must pass through intermediate power modes, these intermediate power modes must also support dynamic transitions.

Figure 3-6 shows an alternate view of the P-Channel PPU power mode transitions to illustrate the intermediate power modes that might be required.

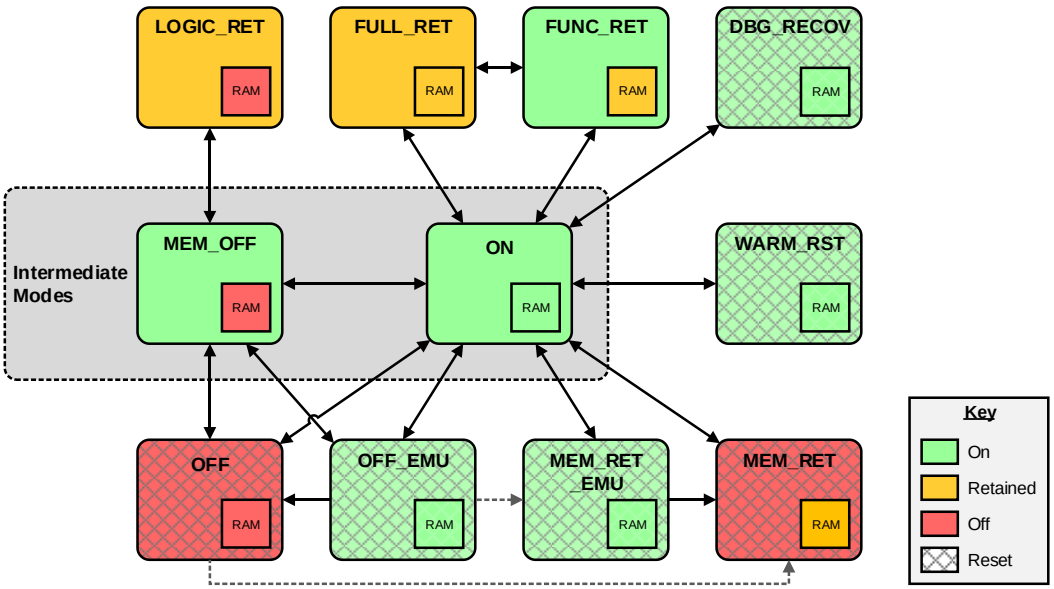

Figure 3-6 P-Channel PPU power mode transitions

While in transition to an intermediate mode the request conditions might change. Therefore, upon reaching an intermediate power mode the PPU must consider the current power mode, programmed policy, and **DEVACTIVE** inputs before making further transitions.

3.4.4 P-Channel Power-Mode Initialization

In OFF, MEM_RET, and DBG_RECOV if the power-on reset is applied, the component P-Channel logic is reset. In these cases, all domain logic state is lost. Therefore, when the reset is de-asserted the PPU is required to initialize the domain to the current mode before transitioning to the requested mode.

This is required as the transition from the current mode to the requested mode, rather than a direct initialization into the requested mode, can have meaning to a component.

For example, consider the following sequence:

- The PPU transitions from ON to MEM_RET. The component is now reset, including any P-Channel control logic.
- The PPU is requested to make a transition back to ON.
    - A transition from MEM_RET to ON, visible to the component, is required as it prevents retained memories from being invalidated.
- The PPU initializes the P-Channel into MEM_RET.
- Once resets are released, and the initialization complete, a further P-Channel transition to ON is carried out.

For power modes where the logic is on, retained or a power on reset is not applied then initialization is not required as knowledge of the current power mode is maintained within the domain.

For full details of the different transition types see *Power Mode Transition Sequences* on page 4-25.

For further information on P-Channel reset and initialization see the *Low-Power Interface Specification: ARM® Q-Channel and P-Channel Interfaces*.

3.5 Operating Mode Transitions

Operating mode transitions are only allowed in certain power mode transitions, these are:

- Between ON and ON.
- Between ON and OFF.
- Between ON and OFF_EMU.
- Between ON and DBG_RECOV.
- Between ON and WARM_RST.
- Between MEM_OFF and OFF.
- Between MEM_OFF and OFF_EMU.
- From OFF to MEM_RET.
- From OFF_EMU to MEM_RET_EMU.

Certain power modes do not have operating mode context, meaning the operating mode setting has no effect on the configuration of the component in this mode. These power modes are:

- OFF.
- OFF_EMU.
- WARM_RST.
- DBG_RECOV.

If the operating mode policy is modified in these power modes, PPU_PWSR and the **PPUHWSTAT** output are updated immediately. No device interface or PCSM handshake occurs. The relevant interrupt events for an operating mode only transition still occur.

Figure 3-7 shows the power mode transitions where operating mode transitions are allowed.

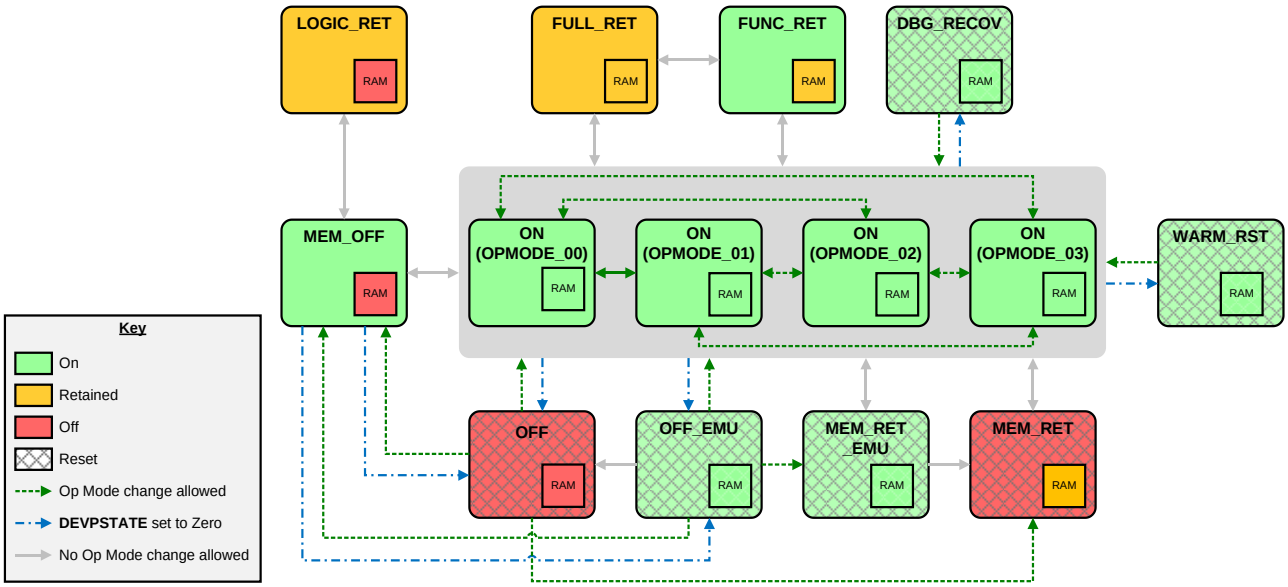

Figure 3-7 Power mode transitions where operating mode transitions are allowed

Transitions from ON to ON are allowed for the purposes of changing the operating mode without changing the power mode. This allows different operating modes during functional operation.

Operating mode changes during power mode transitions, other than ON to ON, happen in two scenarios:

1. When transitioning from a power mode without operating mode context, to one with operating mode context. For example, from OFF to ON:
    - To set the initial component operating mode.
2. When transitioning from a power mode with operating mode context, to one without operating mode context. This causes:
    - The **DEVPSTATE** operating mode bits to be set to a zero.
    - The **PCSMPSTATE** operating mode bits to be set to the highest operating mode.
    - The PPU_PWSR.OP_STATUS field to be updated to reflect the current operating mode policy PPU_PWPR.OP_POLICY, not set to zero.

A transition to a power mode without operating mode context is considered a power and operating mode transition if PPU_PWPR.OP_POLICY does not equal PWSR.OP_STATUS at the start of the transition. Otherwise it is considered a power mode only transition.

Table 3-6 shows operating modes changes allowed on **DEVPSTATE** and **PCSMPSTATE** depending on the power mode transition taking place, and if any restrictions apply.

Transitions between Operating Modes

When moving between operating modes the PPU makes a direct transition to the requested operating mode.

For example, if the PPU is in OPMODE_00 and OPMODE_03 is requested. The PPU transitions directly to OPMODE_03.

Figure 3-8 shows the allowed transitions between operating modes for a PPU which has four operating modes.

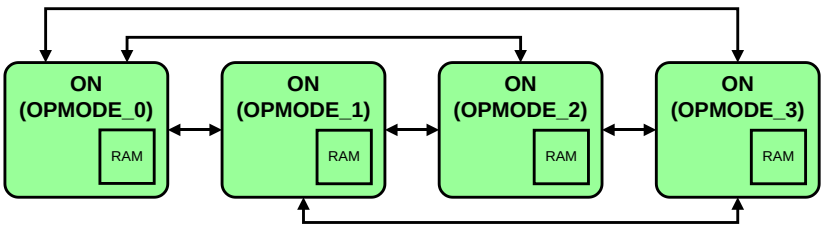

Figure 3-8 – Example of allowed operating mode transitions

Table 3-6 Power mode transitions where DEVPSTATE and PCSMPSTATE operating mode value updates are allowed

Current Power Mode	Power Mode Transition	Operating Mode DEVPSTATE Transition	Operating Mode PCSMPSTATE Transition
ON	ON	Allowed	Allowed
	DBG_RECOV WARM_RST OFF OFF_EMU	Set to zero	Set to highest OPMODE
	MEM_RET_EMU	Not allowed	
	FUNC_RET MEM_OFF FULL_RET MEM_RET	Not allowed	Not allowed
MEM_OFF	DBG_RECOV OFF_EMU OFF	Set to zero	Set to highest OPMODE
	ON LOGIC_RET	Not allowed	Not allowed
FUNC_RET	DBG_RECOV	Set to zero	Set to highest OPMODE
	ON FULL_RET	Not allowed	Not allowed
FULL_RET	DBG_RECOV	Set to zero	Set to highest OPMODE
	ON FUNC_RET	Not allowed	Not allowed
LOGIC_RET	DBG_RECOV	Set to zero	Set to highest OPMODE
	MEM_OFF	Not allowed	Not allowed
DBG_RECOV WARM_RST	ON	Allowed	Allowed
MEM_RET_EMU	DBG_RECOV	Set to zero	Not Required [a]
	ON MEM_RET	Not allowed	Allowed
MEM_RET	DBG_RECOV	Set to zero	Set to highest OPMODE
	ON	Not Allowed	Not Allowed
OFF_EMU	DBG_RECOV	Set to zero	Not Required [a]
	ON MEM_OFF	Allowed	Allowed
	MEM_RET_EMU	Allowed	Not Required [a]
	OFF	Not Allowed	Not Required [a]
OFF	DBG_RECOV	Set to zero	Not Required [a]
	ON MEM_OFF MEM_RET	Allowed	Allowed

[a] The PCSM interface is already set to the highest operating mode, so a transition is not required.

3.6 Operating Mode Transition Rules

This section describes how the PPU initiates operating mode transitions depending on the device interface signalling and software configuration.

Static Operating Mode Change Requests

When PPU_PWPR.OP_DYN_STATUS is 0b0 the operating mode request is based on the programmed operating mode policy.

The operating mode **DEVPACTIVE** inputs do not cause or restrict operating mode transitions.

Dynamic Operating Mode Change Requests

When PPU_PWPR.OP_DYN_STATUS is HIGH the operating mode request is based upon the programmed operating mode policy and the operating mode **DEVPACTIVE** inputs.

The requested operating mode is the highest priority of:

- The operating mode programmed in PPU_PWPR.OP_POLICY.
- The operating mode requested by the **DEVPACTIVE** inputs.

For more information on the operating mode requested by the **DEVPACTIVE** inputs, see *Operating Mode DEVPACTIVE Inputs* on page 3-24.

Operating Mode Transition Behavior

When there is a request to change the operating mode, the PPU transitions to the new operating mode when either:

- The PPU is in ON and there is no power mode request to go to OFF, OFF_EMU, WARM_RST or DBG_RECOV.
    - This results in an ON to ON transition changing the operating mode only.
    - If there is a transition request to any other power mode, this ON to ON transition takes priority.
    - If there is a request to go to OFF or OFF_EMU but the entry delay timer is still counting, then the ON to ON operating mode transition will occur.
- The PPU performs a power mode transition where operating mode transitions are allowed.
    - This results in a simultaneous transition of the power mode and the operating mode.
    - For more information on which power mode transitions allow simultaneous operating mode transitions, see *Operating Mode Transitions* on page 3-21.

When the PPU transitions to power modes without operating mode context it updates the PPU_PWSR.OP_STATUS field with the current operating mode policy, PPU_PWPR.OP_POLICY, even though the operating mode **DEVPSTATE** bits are set to zero.

For more information on power modes do not have operating mode context see *Operating Mode Transitions* on page 3-21.

3.6.1 Operating Mode DEVPACTIVE Inputs

This section describes how the effect of the **DEVPACTIVE** inputs can be modified by software control and how they are used to request different operating modes.

Operating Mode DEVPACTIVE Input Enables

The effect of the operating mode **DEVPACTIVE** inputs on transitions can be enabled and disabled using the PPU.PWCR.OP_DEVACTIVEEN bits.

Disabled operating mode **DEVACTIVE** inputs are always assumed to be LOW for the purposes of transition control.

For further information on these controls see *DEVACTIVE Enables* on page 4-13.

DEVPACTIVE Use Models

Static Use Model

When dynamic operating mode transitions are disabled, PPU_PWPR.OP_DYN_EN is set to 0b0, the **DEVPACTIVE** inputs do not cause or restrict operating mode transitions.

Ladder Use Model

The ladder use model uses the most significant operating mode group **DEVPACTIVE** bit set HIGH to indicate the required operating mode.

The ladder use model supports up to eight operating mode **DEVPACTIVE** inputs and eight operating modes. Each **DEVPACTIVE** is directly related to an operating mode.

The PPU does not define the meaning of the operating modes requested by the **DEVPACTIVE** inputs, only its responses to changes on them.

Table 3-7 shows the operating mode requested based on **DEVPACTIVE** inputs for the ladder use model.

Table 3-7 Ladder use model DEVPACTIVE requests

DEVPACTIVE Bit								Operating Mode Requested
23	22	21	20	19	18	17	16	
1	x	x	x	x	x	x	x	OPMODE_08
0	1	x	x	x	x	x	x	OPMODE_07
0	0	1	x	x	x	x	x	OPMODE_06
0	0	0	1	x	x	x	x	OPMODE_05
0	0	0	0	1	x	x	x	OPMODE_04
0	0	0	0	0	1	x	x	OPMODE_03
0	0	0	0	0	0	1	x	OPMODE_02
0	0	0	0	0	0	0	1	OPMODE_01
0	0	0	0	0	0	0	0	OPMODE_00

Independent Use Model

In the independent use model, each operating mode group **DEVPACTIVE** input bit typically represents a component resource, each of which can be required, or not, regardless of other resources.

The PPU requests an operating mode based on the combination of **DEVPACTIVE** inputs set HIGH, as shown in Table 3-8.

Note

The PPU does not define, or limit, the meaning of operating mode **DEVPACTIVE** inputs other than the operating mode they select.

The independent use model supports up to four operating mode **DEVPACTIVE** inputs and 16 operating modes. As each of the four **DEVPACTIVE** inputs can be used to request a different resource, 16 operating modes are required to represent all the different combinations of the requested resources.

Table 3-8 shows the operating mode requested based on **DEVPACTIVE** inputs.

Table 3-8 Independent Use Model DEVPACTIVE requests

DEVPACTIVE Bit				Operating Mode Requested
19	18	17	16	
1	1	1	1	OPMODE_15
1	1	1	0	OPMODE_14
1	1	0	1	OPMODE_13
1	1	0	0	OPMODE_12
1	0	1	1	OPMODE_11
1	0	1	0	OPMODE_10
1	0	0	1	OPMODE_09
1	0	0	0	OPMODE_08
0	1	1	1	OPMODE_07
0	1	1	0	OPMODE_06
0	1	0	1	OPMODE_05
0	1	0	0	OPMODE_04
0	0	1	1	OPMODE_03
0	0	1	0	OPMODE_02
0	0	0	1	OPMODE_01
0	0	0	0	OPMODE_00

4 Functional Description

This section describes the functional behavior of the PPU. It has the following sections

- *Interfaces* on page 4-2.
- *Power Mode Transition Sequences* on page 4-25.

4.1 Interfaces

The following section describes the software and hardware interfaces of the PPU.

Figure 4-1 shows the interfaces of the PPU.

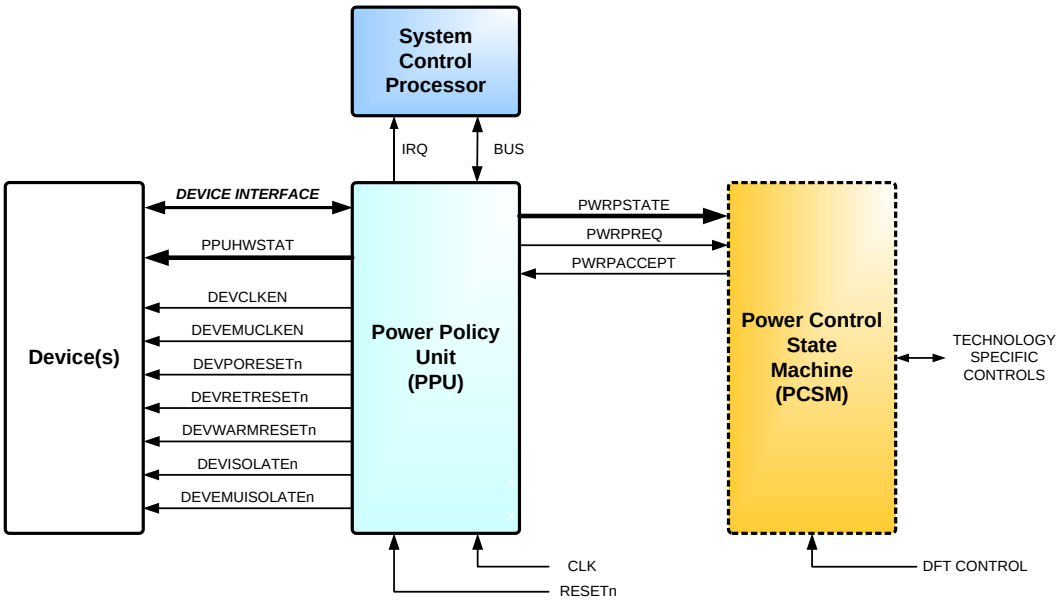

Figure 4-1 PPU interfaces

4.1.1 Software Interface

Software programs the PPU using a memory mapped interface. For more information see the *Programmers Model* on page 5-1.

Power Policy Programming

Software programs the required policy in the *Power Policy Register (PPU_PWPR)*. The PPU completes any ongoing mode transition before the new setting takes effect.

Note

When software modifies the dynamic transition enables, PPU_PWPR.PWR_DYN_EN, or PPU_PWPR.OP_DYN_EN, it must wait until the value is reflected in the corresponding dynamic transition status, PPU_PWSR.PWR_DYN_STATUS or PPU_PWSR.OP_DYN_STATUS, before making further modifications to the dynamic transition enables. Otherwise the PPU behavior is UNPREDICTABLE.

Software writes to the *Power Policy Register (PPU_PWPR)* do not interrupt mode transitions. Transitions only end when one of the following conditions is met:

- The transition completes.
- The request is denied by the PPU device interface.

Policy Programming and Denial

Transition Denial Policy Reversion

If a static transition is denied by the device interface, then:

- The policy reverts to the current mode as shown in the *Power Status Register (PPU_PWSR)*.
- Any dynamic transition enable set to 0b1 during the transition reverts to the value it had when the transition began, as shown in PPU_PWSR.

Table 4-1 shows the policy and dynamic enable values that are reverted on a device interface denial, based on the dynamic enable status, and if the transition updated the power mode, operating mode, or both.

Table 4-1 Policy and dynamic enable reversion on device interface denials

		Policy & Dynamic Enables Reverted on Denial		
PWR_DYN_STATUS	**OP_DYN_STATUS**	**Power Mode Only Transition**	**Operating Mode Only Transition**	**Power and Operating Mode Transition**
0	0	Power and Operating	Power and Operating	Power and Operating
0	1	Power	None	Power
1	0	None	Operating	Operating
1	1	None	None	None

Note

A transition to OFF, OFF_EMU, WARM_RST, or DBG_RECOV is considered a power and operating mode transition if PPU_PWPR.OP_POLICY does not equal PWSR.OP_STATUS at the start of the transition. Otherwise it is considered a power mode only transition.

Denial Interrupt

A static denial interrupt event, if unmasked, occurs when a static transition is denied. Software can use this to perform any required actions, such as to determining why the transition was denied and re-programming the PPU policy.

For further information on this interrupt event see *Static Transition Denial* on page 4-7

For a Q-Channel PPU, the device interface Q-Channels that have denied the request can be determined by reading the *Stored Status Register (PPU_STSR)*, see page 5-8.

Programming Unsupported Policies

If there is an attempt to program an unsupported policy, the *Power Policy Register (PPU_PWPR)* is not updated and, if unmasked, an unsupported policy interrupt event occurs.

Whether a policy is supported is a combination of the policy setting and the dynamic enables. For example, MEM_RET is supported as a static power mode, but not a dynamic power mode. Programming this policy with the power mode dynamic enable set to 0b0 is a valid policy. However, programming this policy with the power mode dynamic enable set to 0b1 is an unsupported policy and would result in the register not being updated and an unsupported policy interrupt event.

For further information on the unsupported policy interrupt event, see *Unsupported Policy* on page 4-9.

Power Mode Emulation Programming

This section describes the use of OFF_EMU and MEM_RET_EMU.

Power Mode Emulated Status

When the PPU is in OFF_EMU and MEM_RET_EMU it is indicated in the PPU_PWSR.PWR_STATUS field.

Emulated Enable Software Control

Software can set the PPU_PMER.EMU_EN bit to use the emulated power modes, OFF_EMU instead of OFF, and MEM_RET_EMU instead of MEM_RET.

For more details on the effects of this bit on power mode transitions see *Power Mode Transition Rules* on page 3-11.

If software changes the EMU_EN bit during a transition it does not take effect until the transition has completed.

Programming OFF_EMU and MEM_RET_EMU Directly

The emulated power modes, OFF_EMU and MEM_RET_EMU, can also be entered by software directly programming the policy value in the PPU_PWPR.PWR_POLICY field.

4.1.2 PPU Interrupt

The PPU has a single, active HIGH, level sensitive, interrupt output.

An interrupt is generated for the following events:

- Static policy transition completion.
    - Full policy transition completion.
    - Power policy transition completion (optional).
    - Operating policy transition completion (optional).
- Static transition acceptance.
- Static transition denial.
- Emulation transition acceptance.
- Emulation transition denial.
- Dynamic transition acceptance.
- Dynamic transition denial.
- Unsupported policy.
- Input Edge.
- Locked (optional).

When an event occurs and it is not masked it sets the corresponding bit in the *Interrupt Status Register (PPU_ISR)* or *Additional Interrupt Status Register (PPU_AISR)*.

The interrupt output is HIGH when any bit in the *Interrupt Status Register (PPU_ISR)* or *Additional Interrupt Status Register (PPU_AISR)* is set.

Each event can be independently masked. When an event is masked, it does not set the corresponding bit in the interrupt status registers.

If an event is cleared in the same cycle that it re-occurs the interrupt status stays HIGH, so the re-occurrence is observed.

Static Policy Transition Completion

This section describes the policy transition completion interrupt events:

- Full policy transition completion.
- Power policy transition completion (optional).
- Operating policy transition completion (optional).

Note

The power policy and operating policy transition completion events are optional. They are not supported when operating modes are not supported. Support for these interrupt events can be determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

These events are used to inform software when the last static policy programmed is reached, they differ on the policy fields that are matched. These events occur when the specified programmed policy is matched, even when this takes more than one transition.

For these events:

- OFF_EMU and OFF are considered equivalent.
- MEM_RET_EMU and MEM_RET are considered equivalent.

Full Policy Transition Completion

The full policy transition completion events occur when the power and operating mode matches the programmed policies, in any of the following conditions:

- Following the write of a supported policy to PPU_PWPR when the PPU is not in transition, and both PPU_PWPR.PWR_DYN_EN and the PPU_PWPR.OP_DYN_EN are set to 0b0.
- An accepted transition completes with both PPU_PWSR.PWR_DYN_STATUS and PPU_PWSR.OP_DYN_STATUS set to 0b0, except for transitions from:
    - OFF_EMU to OFF.
    - MEM_RET_EMU to MEM_RET.
- A denied transition completes with both PPU_PWSR.PWR_DYN_STATUS and PPU_PWSR.OP_DYN_STATUS set to 0b0 and the following conditions are met.
    - PPU_PWPR is written during the transition.
    - The last power and operating policies written match the current power and operating modes.
- Any transition completes with PPU_PWSR.PWR_DYN_STATUS or PPU_PWSR.OP_DYN_STATUS set to 0b1, but both PPU_PWPR.PWR_DYN_EN and PPU_PWPR.OP_DYN_EN are set to 0b0.
    - Meaning a power and operating mode static policy has been programmed during the transition.

This event is unmasked by default.

Note

When operating modes are not supported the behavior is the same as when both PPU_PWPR.OP_DYN_EN, and PPU_PWSR.OP_DYN_STATUS are set to 0b0.

Power Policy Transition Completion

The power policy transition completion event occurs when the power mode matches the programmed power policy, in any of the following conditions:

- Following a write to PPU_PWPR when the PPU is not in transition, and PPU_PWPR.PWR_DYN_EN is set to 0b0.
- A power mode, or a power and operating mode accepted transition completes with PPU_PWSR.PWR_DYN_STATUS set to 0b0, except for transitions from:
    - OFF_EMU to OFF.
    - MEM_RET_EMU to MEM_RET.
- A power mode, or a power and operating mode denied transition completes with PPU_PWSR.PWR_DYN_STATUS set to 0b0 and the following conditions are met.
    - PPU_PWPR is written during the transition.
    - The last power policy written matches the current power mode.
- Any power mode, or a power and operating mode transition completes with PPU_PWSR.PWR_DYN_STATUS set to 0b1, but PPU_PWPR.PWR_DYN_EN is set to 0b0.
    - Meaning a static power mode policy has been programmed during the transition.

This event is masked by default.

For example, the PPU is in FULL_RET and a static policy of OFF is programmed, the PPU transitions to ON before moving to OFF. This event occurs when the PPU completes the transition to OFF.

For example, the PPU is performing a static transition from ON to OFF. During this transition, the policy is programmed to ON. The transition is denied and the policy is reverted to ON. The event occurs as the current power mode matches the last policy programmed during the transition.

Operating Policy Transition Completion

The operating policy transition completion event occurs when the operating mode matches the programmed operating mode policy, in any of the following conditions:

- Following a write to PPU_PWPR when the PPU is not in transition, and PPU_PWPR.OP_DYN_EN is set to 0b0.
- An operating mode, or a power and operating mode accepted transition completes with PPU_PWSR.OP_DYN_STATUS set to 0b0, except when the transitions is:
    - From OFF_EMU to OFF.
    - From MEM_RET_EMU to MEM_RET.
- An operating mode, or a power and operating mode denied transition completes with PPU_PWSR.OP_DYN_STATUS set to 0b0 and the following conditions are met.
    - PPU_PWPR is written during the transition.
    - The last operating policy written matches the current operating mode.
- Any operating mode, or power and operating mode transition completes with PPU_PWSR.OP_DYN_STATUS set to 0b1, but PPU_PWPR.OP_DYN_EN is set to 0b0.
    - Meaning a static operating policy has been programmed during the transition.

This event is masked by default.

Static Transition Acceptance

This event occurs when a static transition is accepted and completed. This includes the completion of intermediate transitions.

For example, if the PPU is in FULL_RET and the programmed policy is OFF. When the PPU transitions to ON the event occurs. Then when the PPU transitions to OFF, the event occurs again.

This event is for all completed static transitions except those covered by the emulated transition completion event. For further information see *Emulation Transition Acceptance* on page 4-7.Table 4-2 indicates whether this event occurs in relation to the dynamic transition status and the type of mode transition, operating mode only, power mode only, or both.

Table 4-2 Static transition event occurrences dependent on dynamic status and transition type

Dynamic Transition Status Setting		Static transition event…		
PPU_PWSR. PWR_DYN_STATUS	PPU_PWSR. OP_DYN_STATUS	…for a power mode only transition.	…for an operating mode only transition.	…for a power and operating mode transition.
0	0	Yes	Yes	Yes
0	1	Yes	No	Yes
1	0	No	Yes	Yes
1	1	No	No	No

This event is masked by default.

Note

A transition to OFF, OFF_EMU, WARM_RST, or DBG_RECOV is considered a power and operating mode transition if PPU_PWPR.OP_POLICY does not equal PWSR.OP_STATUS at the start of the transition. Otherwise it is considered a power mode only transition.

Static Transition Denial

This event occurs when the device interface denies a static transition. This includes denials of intermediate transitions.

For example, if the PPU is in FULL_RET and the programmed policy is OFF, the PPU transitions to ON, before moving to OFF. A denial on either of these transitions is an occurrence of the event.

This event is for all denied static transitions except those covered by the emulated transition denial event. For more information see *Emulation Transition Denial* on page 4-8.Table 4-2 indicates whether this event occurs in relation to the dynamic transition status and the type of mode transition, operating mode only, power mode only, or both.

This event is unmasked by default.

Emulation Transition Acceptance

This event occurs when the following power mode transitions are accepted and completed:

- OFF_EMU to OFF.
- MEM_RET_EMU to MEM_RET.

This event is masked by default.

When both OFF_EMU and MEM_RET_EMU are not supported this event is not present.

Emulation Transition Denial

This event occurs when the following power mode transitions are denied:

- OFF_EMU to OFF.
- MEM_RET_EMU to MEM_RET.

This event is masked by default.

For a Q-Channel PPU, or when both OFF_EMU and MEM_RET_EMU are not supported, this event is not present.

Dynamic Transition Acceptance

This event occurs when a dynamic transition is accepted and completed.

This event is for all completed dynamic transitions except those covered by the emulated transition completion event. For more information see Emulation Transition on page 4-7.

Table 4-3 indicates whether this event occurs in relation to the dynamic transition status and the type of mode transition, operating mode only, power mode only, or both.

Table 4-3 Dynamic transition event occurrences dependent on dynamic status and transition type

Dynamic Transition Status Setting		Dynamic transition event…		
PPU_PWSR. PWR_DYN_STATUS	PPU_PWSR. OP_DYN_STATUS	…for a power mode only transition.	…for an operating mode only transition.	…for a power and operating mode transition.
0	0	No	No	No
0	1	No	Yes	Yes
1	0	Yes	No	Yes
1	1	Yes	Yes	Yes

Note

A transition to OFF, OFF_EMU, WARM_RST, or DBG_RECOV is considered a power and operating mode transition if PPU_PWPR.OP_POLICY does not equal PWSR.OP_STATUS at the start of the transition. Otherwise it is considered a power mode only transition.

This event is masked by default.

Dynamic Transition Denial

This event occurs when the device interface denies a dynamic transition.

This event is for all denied dynamic transitions except those covered by the emulated transition denial event. For more information see *Emulation Transition Denial* on page 4-8.

Table 4-3 indicates whether this event occurs in relation to the dynamic transition status and the type of mode transition, operating mode only, power mode only, or both.

This event is masked by default.

Unsupported Policy

This event occurs when an unsupported policy value is programmed.

For further information on programming unsupported policies see *Programming Unsupported Policies* on page 4-3.

Note

This interrupt is valid for policies unsupported by this configuration of PPU and for reserved policy values.

This event is unmasked by default.

Input Edge

This event occurs when **DEVACTIVE** inputs configured for this event change. The event can be separately configured for each **DEVACTIVE** input bit.

This operates directly on the **DEVACTIVE** inputs before they are affected by the **DEVACTIVE** enables and the emulation enable, for more on these see *Power Mode DEVACTIVE Input Enables* on page 3-11, *Operating Mode DEVPACTIVE Input Enables* on page 3-24 and *Power Mode DEVACTIVE Inputs and Emulation Enable* on page 3-11.

Each input can be configured so the event:

- Is masked.
- Occurs on the rising edge.
- Occurs on the falling edge.
- Occurs on both edges.

For more information see the *Input Edge Sensitivity Register (PPU_IESR)* on page 5-15.

The status of each **DEVACTIVE** input can be read. See the *Device Interface Input Current Status Register (PPU_DISR)* on page 5-6.

By default, all input edge events are masked.

Note

For a P-Channel PPU, there is no input edge event support for **DEVPACTIVE[0]**. This input represents OFF, and this is typically not driven by components.

Locked

This event is used to inform software that the PPU has become locked in a power mode.

Note

The locked interrupt event is optional. Whether it is supported is determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

This event occurs when the locked status bit, PPU_PWSR.LOCK_STATUS is set to 0x1:

This event is masked by default.

4.1.3 Device Interface

The interface from the PPU to the device consists of one or more LPIs. The device interface is either:

- Between one and eight Q-Channels.
- A single P-Channel.

The interface is chosen according to the devices being connected.

For more information on these interfaces see the *Low Power Interface Specification ARM Q-Channel and P-Channel Interfaces.*

Device Q-Channel Interface

The PPU is configured with up to eight device interface Q-Channels. Each one is connected to a different device Q-Channel to ensure that the device is ready to enter the required power mode.

The PPU ensures that all Q-Channels are in *Q_STOPPED* before a lower priority power mode is entered. If any Q-Channel issues a deny response then this is considered a denial of entry to the power mode, and all Q-Channels are returned to *Q_RUN*, no power mode change takes place.

A device interface Q-Channel consists of the following signals:

- **DEVQACTIVE**.
- **DEVQREQn**.
- **DEVQACCEPTn**.
- **DEVQDENY**.

Note

This interface can be connected to a device with an AXI LPI Interface, subject to specific limitations. For more information see *AXI LPI Support* on page 8-2.

For all policies other than ON and WARM_RST, all enabled **DEVQACTIVE** inputs must be LOW before a device interface quiescence entry request is made. The interface then follows the Q-Channel protocol.

Figure 4-2 shows a device exiting quiescence between t0 and t3, and then entering quiescence between t3 and t6.

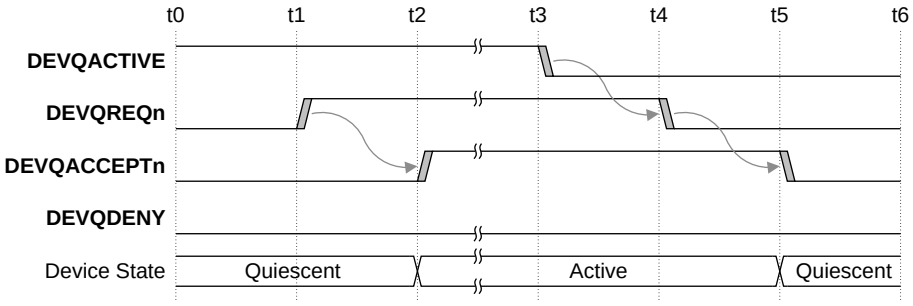

Figure 4-2 Device interface: Q-Channel acceptance

Figure 4-3 shows an example of a transition request that is denied by the device. The device stays active and available for use throughout the entire period.

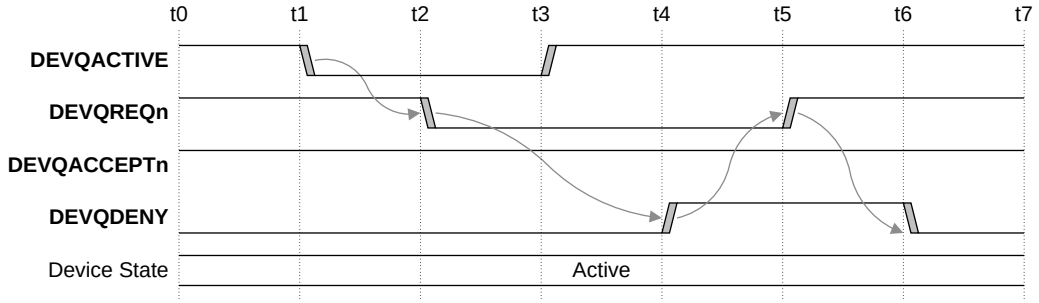

Figure 4-3 Device interface: Q-Channel denial

Figure 4-4 shows an example of a dynamic transition where the entry and exit are initiated by **DEVQACTIVE**.

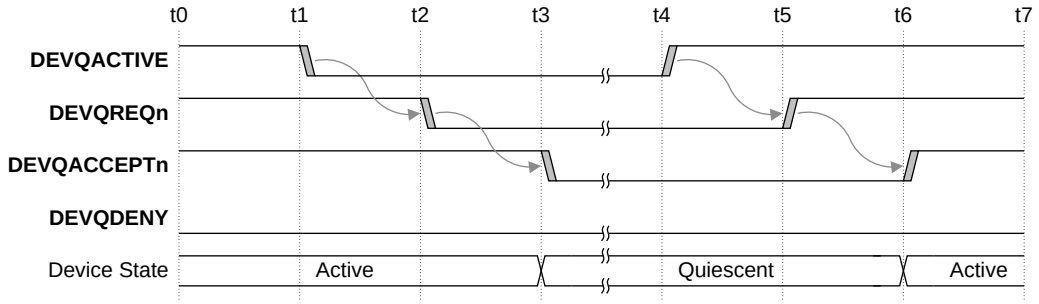

Figure 4-4 Device interface: Q-Channel dynamic entry and exit

Note

For clarity, the diagram only shows a single **DEVQACTIVE** input, but this represents an aggregation of all the **DEVQACTIVE** inputs.

The functions of each device interface Q-Channel can be enabled and disabled by software using the PPU_PWCR.PWR_DEVACTIVEEN, and PPU_PWCR DEVREQEN fields.

Device P-Channel Interface

A P-Channel PPU has a single device interface P-Channel. This ensures that the domain is ready to enter the requested power mode.

The device interface P-Channel consists of the following signals:

- **DEVPACTIVE[23:0].**
- **DEVPSTATE[7:0].**
- **DEVPREQ.**
- **DEVPACCEPT.**
- **DEVPDENY.**

Note

The widths for the **DEVPACTIVE** and **DEVPSTATE** signals are the maximum signal widths, they can be reduced if not all modes are supported.

DEVPACTIVE has multiple bits, with each bit indicating the requirement for a mode. For example, one bit indicates the requirement for ON, while another indicates the requirement for FUNC_RET.

Power modes are represented on **DEVPACTIVE[10:0]**, operating modes are represented on **DEVPACTIVE[23:16]**. **DEVPACTIVE[15:11]** are reserved.

The interface follows the P-Channel protocol, as illustrated in Figure 4-5

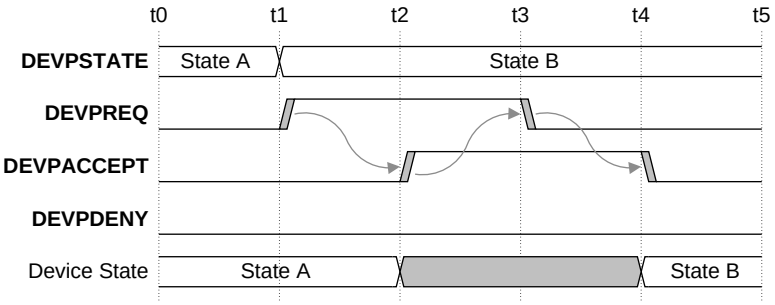

Figure 4-5 Device interface: P-Channel acceptance

Figure 4-6 shows an example of a transition request that is denied by a device. The device stays in the previous mode throughout the entire period.

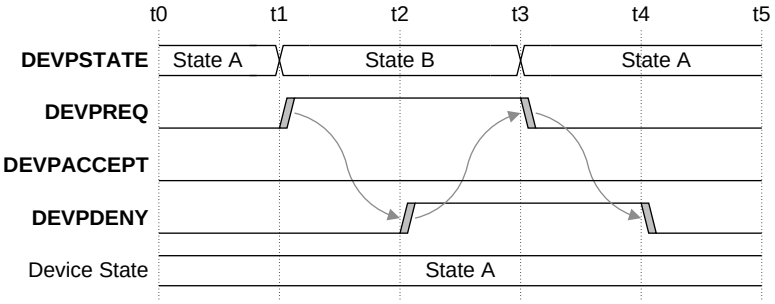

Figure 4-6 Device interface: P-Channel denial

Figure 4-7 shows an example dynamic transition scenario where the entry and exit are initiated by **DEVPACTIVE**.

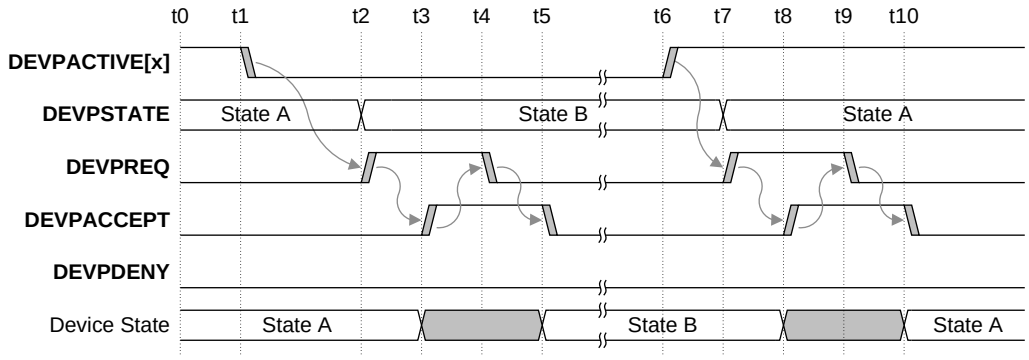

Figure 4-7 Device interface : P-Channel dynamic transition

The functions of the device interface P-Channel can be enabled and disabled by software using the PPU_PWCR.PWR_DEVACTIVEEN, PPU_PWCR.OP_DEVACTIVEEN, and PPU_PWCR DEVREQEN fields.

DEVPSTATE Early Assertion

The P-Channel can be optionally configured to set the **DEVPSTATE** value zero or more PPU clock cycles before **DEVPREQ** goes HIGH. This allows an additional timing margin, DEV_PREQ_DLY, between these signals when they cross from the PPU to the device.

The value of DEV_PREQ_DLY is also applied when the **DEVPSTATE** value is reverted because a device denied a transition request. In this case, the delay applies between the **DEVPSTATE** value being changed and **DEVPREQ** being set LOW.

When DEV_PREQ_DLY is zero **DEVPSTATE** and **DEVPREQ** change on the same clock edge.

Figure 4-8 shows the use of the DEV_PREQ_DLY value when asserting **DEVPREQ** HIGH.

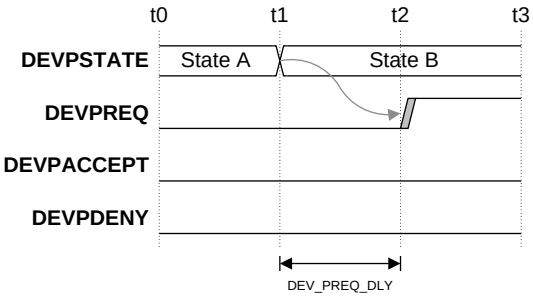

Figure 4-8 Use of DEV_PREQ_DLY when asserting DEVPREQ

Figure 4-9 shows the use of the DEV_PREQ_DLY value when de-asserting **DEVPREQ** following a device denial.

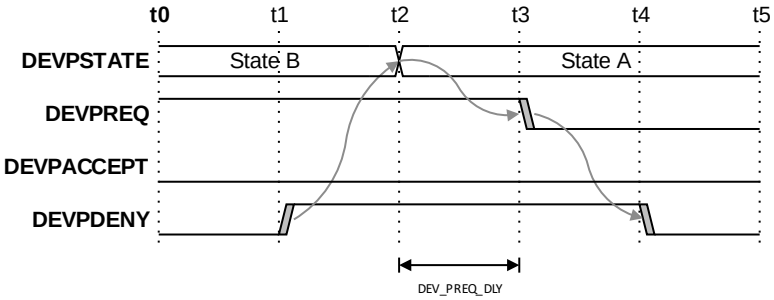

Figure 4-9 Use of DEV_PREQ_DLY when de-asserting DEVPREQ following a device denial

DEVACTIVE Enables

Software can control the **DEVACTIVE** signals that are used in the PPU to initiate transitions between power modes. There is a separate control for each **DEVACTIVE** bit. All **DEVACTIVE** inputs are enabled at reset.

These enables have no effect on the generation of interrupts. If an interrupt for a **DEVACTIVE** is unmasked it still generates an interrupt event when the input changes. This allows **DEVACTIVE** changes to be monitored by software.

Device Interface Channel Enables

Software can enable and disable the device interface channels that are used when the PPU transitions between modes.

Each interface can be disabled individually for all transitions, or all channels can be disabled for certain transitions.

When a device interface channel is disabled:

- **P-Channel: DEVPREQ** is driven permanently LOW and **DEVPSTATE** does not update.
- **Q-Channel: DEVQREQn** is driven permanently HIGH.

When a device interface channel is disabled, the corresponding **DEVACCEPT** and **DEVDENY** inputs are ignored and transitions continue without issuing requests or waiting for responses on the disabled channels.

Individual Channel Enables

Each of the Q-Channels or the P-Channel can be enabled and disabled individually for all transitions.

The PPU_PWCR.DEVREQEN field is used for this control and has the following bits, dependent on the PPU type:

- **P-Channel PPU:** One bit to control the device interface P-Channel.
- **Q-Channel PPU:** One bit per device interface Q-Channel.

Note

Before software programs any DEVREQEN bits it must configure the PPU for static transitions and ensure the requested power mode has been reached, this means that no further transitions can occur, otherwise behavior is UNPREDICTABLE.

Additionally, when using a Q-Channel PPU, it must be in ON before software enables any previously disabled Q-Channels, otherwise behavior is UNPREDICTABLE.

All PPU device interface channels are enabled at reset.

Transition Specific Device Interface Channel Enables

The device interface P-Channel, or all Q-Channels, can be enabled or disabled for transitions between ON and WARM_RST. This allows components without specific WARM_RST support to be warm reset using the PPU.

The PPU_PTCR.WARM_RST_DEVREQEN bit is used for this control. The default value of this bit is configurable, see *Transition Control Parameters* on page 7-3.

Table 4-4 shows the effects of the software settings on device interface handshakes for transitions between ON and WARM_RST.

Table 4-4 Device interface enable software settings effects on WARM_RST transitions

Transition	PPU_PWCR. DEVREQEN[x]	PPU_PTCR. WARM_RST_DEVREQEN	Device Interface[x] Handshake
ON to WARM_RST WARM_RST to ON	0	X	No
	1	0	No
	1	1	Yes

Power Mode Entry Delay Timers

The power mode entry delay function allows hysteresis to be applied for transitions to lower priority power modes. This is useful where the penalty of exiting the power mode might be high so adding hysteresis might be beneficial before entering a mode.

Note

Power mode entry delay timer functionality is an optional feature. Whether it is supported is determined by reading *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

The power mode entry delay timers are used to control the time between the conditions being met to transition to a power mode and the transition occurring. If, while the entry delay timer is counting, conditions change such that the transition being counted should no longer take place, then the counter is reset.

These delays apply for transitions to a lower priority power mode, except for the following transitions:

- From WARM_RST to ON.
- From DBG_RECOV to ON.
- From MEM_RET_EMU to MEM_RET
- From OFF_EMU to OFF.

Transitions to higher priority power modes always happen as soon as the conditions are met.

There is a separate entry delay value for each power mode and it is applied when the conditions are met for that power mode to be entered.

Transitions to OFF_EMU use the entry-delay value for OFF.

Transitions to MEM_RET_EMU use the MEM_RET power mode entry delay value.

Some example behaviors are:

- Example 1: Q-Channel PPU, in the ON mode, has a programmed policy of FUNC_RET. When all the enabled power mode **DEVACTIVE** inputs go LOW then the conditions for the transition are met and the entry delay timer begins counting. If these conditions remain the same until the timer expires, the transition to FUNC_RET begins.
- Example 2: P-Channel PPU, in the ON mode, has a programmed policy of dynamic OFF. The ON **DEVACTIVE** goes LOW but the MEM_OFF **DEVACTIVE** remains HIGH. This generates an internal request to the MEM_OFF mode and the delay timer starts with the MEM_OFF value. If these conditions remain the same until the timer expires, the transition to MEM_OFF begins.

Figure 4-10 shows example behaviour of the delay timer.

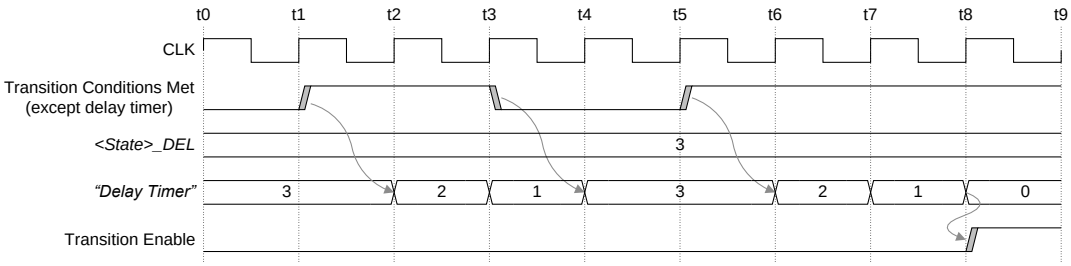

Figure 4-10 Delay timer example

If the delay timer value is re-programmed while the delay timer is counting, for example between t1 to t3 and t5 to t9 in Figure 4-10, the timer delay continues to use the previous programmed value until the count is restarted.

4.1.4 PPUHWSTAT

The **PPUHWSTAT** output indicates the current power and operating mode. This signal can be used by other hardware elements that depend upon the mode of the PPU, for example, other PPUs. This capability can be used to manage power-domain dependencies in hardware.

The **PPUHWSTAT** field is one-hot encoded to simplify its use in hardware qualification of mode status. This is different from the programmed policy, and the device and PCSM P-Channel **PSTATE** values.

Table 3-2 on page 3-6 shows the power mode encoding for **PPUHWSTAT**.

Table 3-3 on page 3-7 shows the operating mode encoding for **PPUHWSTAT**.

4.1.5 Device Controls

The device controls consist of the clock enable, isolation, and reset controls. These are the **DEV*CLKEN**, **DEV*ISOLATEn**, and **DEV*RESETn** signals.

Typically, when moving to a higher priority power mode the signals are sequenced after the PCSM handshake has completed and before the device interface handshake is started:

1. Isolation de-asserted.
2. Clock enables asserted.
3. Resets de-asserted.

Typically, when moving to a lower priority power mode, the signals are sequenced after the device interface handshake, and before the PCSM handshake.

1. Clock enables de-asserted.
2. Isolation enabled.
3. Resets asserted.

Example device control timing diagrams are given in *Device Control Examples* on page 4-20.

Example power mode transition timing diagrams are given in *Power Mode Transition Sequences* on page 4-25.

Device Control Delays

The number of clock cycles between different signals changing is either statically configured, or optionally, runtime-configured by software. See *Device Control Delay Software Configurability* on page 4-17.

This feature facilitates support for devices that have requirements on the timing between certain device control signals. Table 4-5 shows the different configurable parameters.

Table 4-5 Device control delay parameters

Parameter	Description
ISO_CLKEN_DLY	The minimum number of PPU clock cycles between **DEV*ISOLATEn** de-assertion and **DEVCLKEN** assertion is ISO_CLKEN_DLY + 1.
CLKEN_RST_DLY	The minimum number of PPU clock cycles between **DEVCLKEN** assertion and **DEV*RESETn** de-assertion is CLKEN_RST_DLY + 1.
RST_HWSTAT_DLY	The minimum number of PPU clock cycles between **DEV*RESETn** de-assertion and **PPUHWSTAT** updating is RST_HWSTAT_DLY + 1. This delay is also dependant on the completion of the device interface.
CLKEN_ISO_DLY	The minimum number of PPU clock cycles between **DEVCLKEN** de-assertion and **DEV*ISOLATEn** assertion is CLKEN_ISO_DLY + 1.
ISO_RST_DLY	The minimum number of PPU clock cycles between **DEV*ISOLATEn** assertion and **DEV*RESETn** assertion is ISO_RST_DLY + 1.

Valid values for each parameter are in the range 0-255.

These values apply whenever these signals change, this includes when entering a mode or when reverting a mode entry due to a device interface request denial. For more information on where these values apply see *Device Control Examples* on page 4-20 and *Power Mode Transition Sequences* on page 4-25.

When calculating these settings allowances must be made for differences in clock frequencies between the PPU and the controlled device and any other delays such as re-synchronization.

For example, if CLKEN_RST_DLY is used to guarantee that the device reset is LOW for several device clock cycles, the clock enable re-synchronization requirements must be considered. If the

clock enable is re-synchronized the clock might not be running as soon as the clock enable is asserted from the PPU, so a larger delay period might need to be used. In addition, the differences in clock frequency between the PPU and the device must be considered when setting the value.

Device Control Delay Software Configurability

The PPU can support runtime software configurability of the device control delays. Support for this feature is optional. Software can determine whether this feature is supported by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

The default values of these registers can be configured, see *Device Control Delay Parameters* on page 7-3.

Device Clock Enables

The clock enables are used to ensure clocks to the domain are stopped before entering certain power modes. All clock enables are active HIGH.

Table 4-6 lists the clock enable outputs.

Table 4-6 Clock enable descriptions

Signal	Description
DEVCLKEN	Used to control clocks that must be stopped for power modes where the domain logic is retained or off. This also includes OFF_EMU and MEM_RET_EMU as this emulates the functionality of OFF and MEM_RET. Clocks that need to be running in these modes use the **DEVEMUCLKEN** control.
DEVEMUCLKEN	Used to control clocks that must not be stopped in OFF_EMU and MEM_RET_EMU. For example, clocks that need to run to allow debug accesses to continue for a processor in OFF_EMU.

Table 4-7 shows the clock enable output levels for each power mode.

Table 4-7 Clock enable behavior for each power mode

Power Mode	DEVCLKEN	DEVEMUCLKEN
DBG_RECOV WARM_RST ON FUNC_RET MEM_OFF	1	1
OFF_EMU MEM_RET_EMU	0	1
FULL_RET LOGIC_RET MEM_RET OFF	0	0

Note

These signals must be re-synchronized if the clocks being controlled are asynchronous to the PPU to ensure reliable enabling and disabling of clocks.

Device Resets

There are several reset outputs from the PPU that are asserted in different circumstances. All resets are active LOW.

Table 4-8 lists the reset outputs.

Table 4-8 Reset descriptions

Signal	Description
DEVWARMRESETn	Warm reset for non-retention registers, connected to registers that are not to be retained
DEVRETRESETn	Warm reset for retention registers, connected to registers that are to be retained
DEVPORESETn	Power on reset for all registers that are not reset with either the **DEVWARMRESET** or **DEVRETRESET** signals. This reset is asserted in OFF and MEM_RET only.

Table 4-9 shows the reset signals levels for each power mode.

Table 4-9 Reset behavior for each power mode

Power Mode	DEVPORESETn	DEVWARMRESETn	DEVRETRESETn
ON MEM_OFF FUNC_RET	1	1	1
FULL_RET LOGIC_RET	1	0	1
DBG_RECOV	See *DBG_RECOV and DEVPORESETn* on page 4-18	0	0
WARM_RST MEM_RET_EMU OFF_EMU	1	0	0
MEM_RET OFF	0	0	0

DBG_RECOV and DEVPORESETn

Software can configure if the **DEVPORESETn** is applied when the PPU is in DBG_RECOV. It does this by configuring the PPU_PTCR.DBG_RECOV_PORST_EN. This bit also controls the entry and exit sequences of DBG_RECOV.

Note

This bit should not be modified when the PPU is in DBG_RECOV, or if a PPU transition is ongoing, if it is PPU behavior is UNPREDICTABLE.

For more information on these sequences see *Transitions to and from DBG_RECOV* on page 4-35.

Component Resets

It is recommended that the component logic that controls the P-Channel should be on the deepest reset.

Where a component power domain has a single power on reset the P-Channel control logic for the domain is recommended to be on this reset. This is the case even if the P-Channel control logic resides outside of the power domain it is controlling.

Where a component power domain has multiple resets, such as a warm reset and a power on reset, the P-Channel control logic is recommended to be on the power on reset.

This ensures the components P-Channel logic is only reset when the power domain logic is powered off.

For components that support DBG_RECOV, software can configure if **DEVPORESETn** is asserted whilst in this power mode. The following statements assume the component P-Channel logic is reset by **DEVPORESETn**.

When **DEVPORESETn** is asserted in DBG_RECOV the component is not required to support transitioning to this power mode, only to be initialized into it, and exit from it. All logic is reset, therefore state required to be preserved through the reset must reside in RAM.

When **DEVPORESETn** is not asserted in DBG_RECOV the component must support initialization and entry into, and exit from, this power mode. Not all logic is reset, therefore state required to be preserved can reside in registers reset by **DEVPORESETn**, or in RAM.

Device Isolation Control

There are two isolation control outputs from the PPU that are asserted in different circumstances. All isolation control outputs are active LOW.

Table 4-10 lists the isolation control outputs.

Table 4-10 Isolation signal descriptions

Signal	Description
DEVISOLATEn	Connected to isolation cells that isolate in FULL_RET, LOGIC_RET, MEM_RET_EMU, MEM_RET, OFF_EMU, and OFF.
DEVEMUISOLATEn	Used to control isolation that must not be isolated in OFF_EMU and MEM_RET_EMU. For example, outputs that need to remain un-isolated to allow debug accesses to continue for a processor in either OFF_EMU or MEM_RET_EMU.

Table 4-11 shows the isolation signal levels for each power mode.

Table 4-11 Isolation signal behavior for each power modes

Power Mode	DEVISOLATEn	DEVEMUISOLATEn
DBG_RECOV WARM_RST ON FUNC_RET MEM_OFF	1	1
MEM_RET_EMU OFF_EMU	0	1
FULL_RET LOGIC_RET MEM_RET OFF	0	0

Device Control Examples

Figure 4-11 shows the sequence when transitioning from ON to OFF.

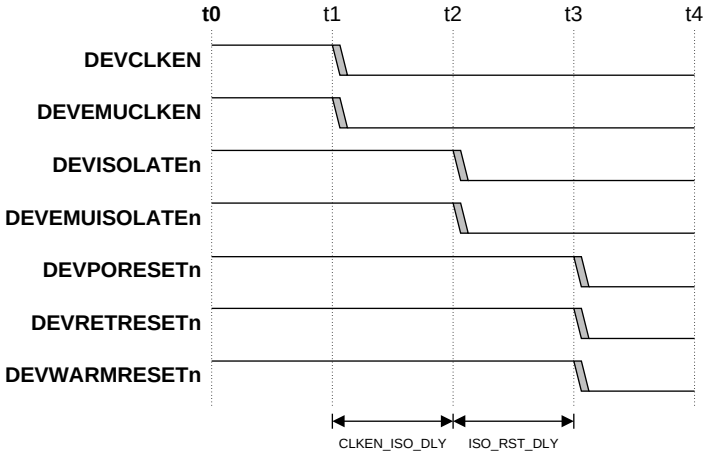

Figure 4-11 Device control transition from ON to OFF

Figure 4-12 shows the sequence when transitioning from OFF to ON.

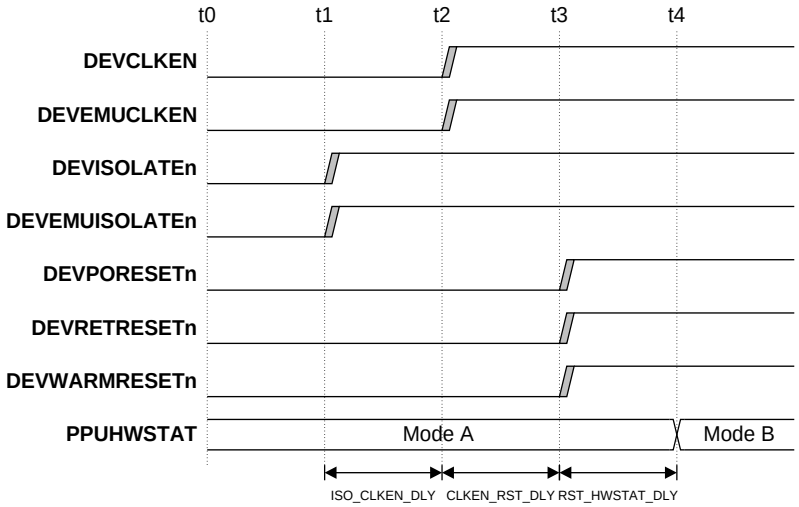

Figure 4-12 Device control transition from OFF to ON

Figure 4-13 shows a transition from ON to OFF_EMU, between t0 and t3. This is followed by the PPU_PMER.EMU_EN being set to 0b0 at t4, resulting in a transition to OFF, between t4 and t7.

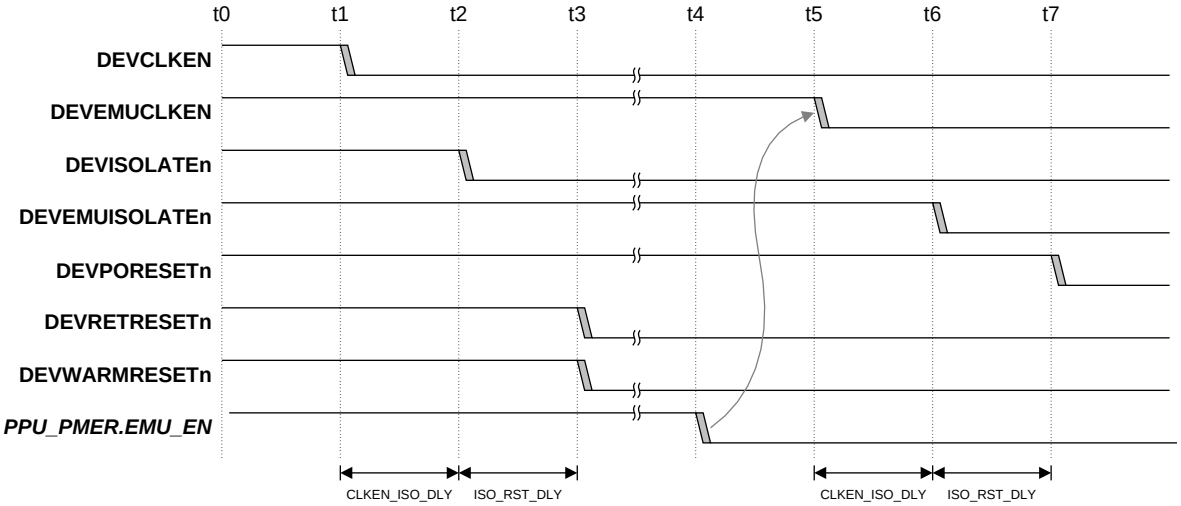

Figure 4-13 Device control transition from ON to OFF_EMU, then to OFF

Figure 4-14 shows a transition from ON to LOGIC_RET or ON to FULL_RET.

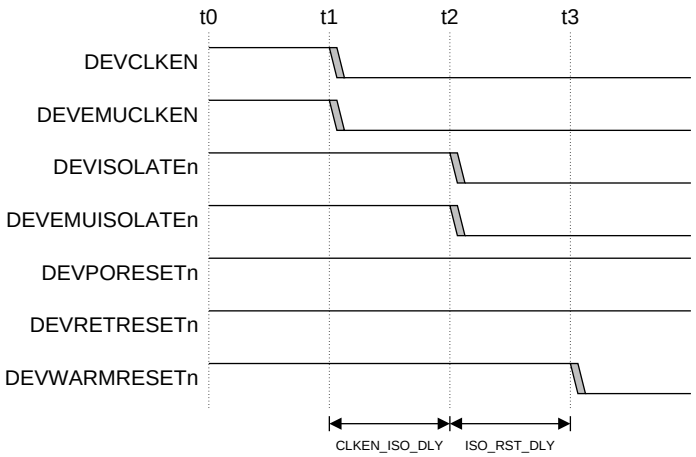

Figure 4-14 Device control transition from ON to LOGIC_RET or FULL_RET

4.1.6 PPU to PCSM Interface

PCSM P-Channel

When the PPU requires a transition of technology-specific controls it requests the change on the PCSM P-Channel. Examples of technology-specific controls include power domain switches and retention controls.

This PCSM P-Channel is compliant with the P-Channel protocol described in the *Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces*. It has no **PDENY** or **PACTIVE** signals.

The PCSM P-Channel consists of the following signals:

- **PCSMPSTATE[15:0].**
- **PCSMPREQ.**
- **PCSMPACCEPT.**

An example of a P-Channel protocol transition is illustrated in Figure 4-15.

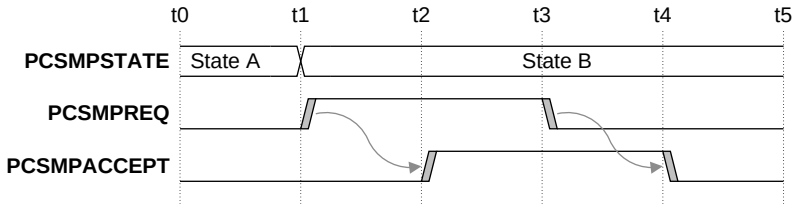

Figure 4-15 PPU to PCSM P-Channel transition protocol

The power transition in the PCSM must happen between t1 and t4 in Figure 4-15, before the P-Channel transition is completed by the PCSM.

PCSM P-Channel Reset

The **PCSMPSTATE** output is reset to the default power mode of the PPU.

If the default state of the PPU is ON, when the PPU is released from reset it immediately makes a transition to ON. This ensures that the reset value is sampled by the PCSM before any other transition is allowed on the interface.

If the default state of the PPU is OFF, when the PPU is released from reset it is IMPLEMENTATION DEFINED if a P-Channel transition is made to the PCSM.

Figure 4-16 shows the PCSM P-Channel behavior at reset de-assertion when a PCSM P-Channel transition occurs.

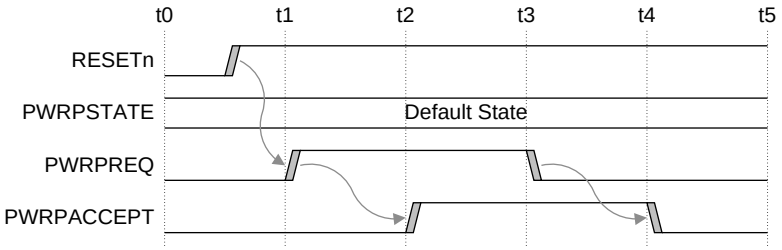

Figure 4-16 PCSM P-Channel at PPU Reset

If the initial power mode of the PPU is set to MEM_RET by the **PCSMMODESTAT** input, this is not reflected on the **PCSMPSTATE** output. **PCSMPSTATE** will be reset as stated above and does not change until there is a transition on the PCSM P-Channel.

For further information on the **PCSMMODESTAT** input see *PCSMMODESTAT* on page 4-24.

Note

Depending on the reset timings between the PPU and PCSM, the PCSM might sample the **PCSMPSTATE** value at reset release, before sampling the **PCSMPREQ** HIGH. When the PCSM samples **PCSMPREQ** HIGH it might see this as a further transition request to the current mode. Whenever the PCSM sees a transition request to its current mode it must accept the request and take no further action.

PCSMPSTATE Enumeration

The **PCSMPSTATE** enumeration of the PCSM P-Channel is equivalent to that of the device interface, concatenated with the value from the RAM Configuration Register for the mode being entered. This gives the PCSM information on how the RAM retention must be configured.

The **PCSMPSTATE** enumeration is shown in Table 4-12.

Table 4-12 PCSMPSTATE signal enumeration

PCSMPSTATE Bit	PCSMPSTATE Value Source
[15:8]	RAM Retain Value, depending on the power mode being requested: **FUNC_RET**: FUNC_RET_RAM_CFG, see the *Functional Retention RAM Configuration Register (PPU_FUNRR)* on page 5-18. **FULL_RET**: FULL_RET_RAM_CFG, see the *Full Retention RAM Configuration Register (PPU_FULRR)* on page 5-18. **MEM_RET**: MET_RET_RAM_CFG, see the *Memory Retention RAM Configuration Register (PPU_MEMRR)* on page 5-19. For all other transitions these bits are set to 0b00000000.
[7:4] [a]	The **PSTATE** value related to the operating mode being requested. For enumeration information see Table 3-3 on page 3-7. If the PPU is configured as a Q-Channel PPU, or operating modes are not supported, this field is reserved and is always set to 0b0000.
[3:0]	The **PSTATE** value related to the power mode being requested. For enumeration information see Table 3-2 on page 3-6.

[a] If the OPMODE_PCSM_SPT_CFG parameter is set to 0, meaning PCSM transitions for operating mode only transitions are disabled, these outputs are tied LOW. For further information see *Operating Mode PCSM Transition Configuration* on page 7-7.

PCSMPSTATE Early Assertion

The P-Channel can be configured to set the **PCSMPSTATE** value a number of PPU clock cycles before **PCSMPREQ** goes HIGH. This allows an additional timing margin, PCSM_PREQ_DELAY, between these signals when they cross from the PPU to the PCSM. When PCSM_PREQ_DLY is zero, **PCSMPSTATE** and **PCSMPREQ** change on the same clock edge.

Figure 4-17 shows the use of the PCSM_PREQ_DLY value when asserting **PCSMPREQ** HIGH.

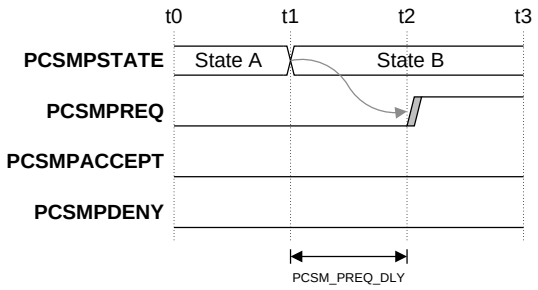

Figure 4-17 – Use of PCSM_PREQ_DLY when asserting PCSMPREQ

PCSMMODESTAT

The **PCSMMODESTAT** input is an optional signal that indicates at PPU reset exit the current power mode of the PCSM.

This input exists if the PPU is configured to support OFF to MEM_RET transitions, see *OFF to MEM_RET Direct Transition Configuration* on page 7-5. This allows the domain the PPU controls to be in MEM_RET whilst the PPU is reset and powered-off.

When the PPU reset is released it can be re-configured by software from the default of OFF into MEM_RET. The **PCSMMODESTAT** signal provides the same functionality without requiring software programming. If the PCSM is in MEM_RET, then when the PPU exits reset the PPU takes this value from **PCSMMODESTAT** and makes it the current PPU mode.

This input is only sampled by the PPU at reset release. When **PCSMMODESTAT[3:0]** indicates a MEM_RET power mode the PPU does the following:

- Sets the power mode status, PPU_PWSR.PWR_STATUS, to the **PCSMMODESTAT** value.
- Sets the operating mode status, PPU_PWSR.OP_STATUS, to the **PCSMMODESTAT** value.
- Sets the **PPUHWSTAT** to the mode indicated by the **PCSMMODESTAT**.
- If the default power policy, DEF_PWR_POLICY, is OFF, sets PPU_PWPR.PWR_POLICY to MEM_RET.
- Sets the operating mode policy, PPU_PWPR.OP_POLICY, to the **PCSMMODESTAT** value.

Otherwise all these settings take their normal default value.

Note

If OPMODE_PCSM_SPT_CFG is set to 0 the operating mode policy and status are not updated as the operating modes are not communicated to the PCSM.

PCSMMODESTAT is encoded with the same values as the device P-Channel **PSTATE** values.

Only the values for OFF and MEM_RET are allowed on **PCSMMODESTAT[3:0]**. Therefore, bits [3:2] can be tied LOW.

All values are allowed on **PCSMMODESTAT[7:4]**. If operational modes are not supported, then these bits can be tied LOW.

Table 4-13 show the effect of **PCSMMODESTAT** on the PPU initialization behavior based on the default power policy, DEF_PWR_POLICY, and the **PCSMMODESTAT** input. Where the **PCSMMODESTAT** is not present the behavior is as if it is set to OFF.

Table 4-13 PCSMMODESTAT effect on initialization behavior

Default Power Policy (DEF_PWR_POLICY)	PCSMMODESTAT	Behaviour
OFF	OFF / Not Present	Initialises into OFF
	MEM_RET	Initialises into MEM_RET
ON	OFF / Not Present	Initialises into ON
	MEM_RET	Initialises into MEM_RET, then transitions to ON

4.2 Power Mode Transition Sequences

This section describes the external signaling for all interfaces during power mode transitions.

Table 4-14 shows the allowed transitions and lists the section that describes the transition type.

Table 4-14 Power mode transition types

Start Power Mode	End Power Mode	Transition Section
OFF	MEM_RET	Transitions from OFF to MEM_RET
	MEM_OFF	Transitions from OFF or MEM_RET, to a Higher Priority Power Mode
	ON	Transitions from OFF or MEM_RET, to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV
OFF_EMU	MEM_RET_EMU	Transitions between OFF_EMU and MEM_RET_EMU
	OFF	Transitions to a Lower Priority Power Mode
	MEM_OFF	Transitions to a Higher Priority Power Mode
	ON	Transitions to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV
MEM_RET	ON	Transitions from OFF or MEM_RET, to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV
MEM_RET_EMU	MEM_RET	Transitions to a Lower Priority Power Mode
	ON	Transitions to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV
LOGIC_RET	MEM_OFF	Transitions to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV
FULL_RET	FUNC_RET	Transitions to a Higher Priority Power Mode
	ON	Transitions to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV
MEM_OFF	OFF	Transitions to a Lower Priority Power Mode
	OFF_EMU	Transitions to a Lower Priority Power Mode
	LOGIC_RET	Transitions to a Lower Priority Power Mode
	ON	Transitions to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV
FUNC_RET	FULL_RET	Transitions to a Lower Priority Power Mode
FUNC_RET	ON	Transitions to a Higher Priority Power Mode
	DBG_RECOV	Transitions to and from DBG_RECOV

Start Power Mode	End Power Mode	Transition Section
ON	OFF	Transitions to a Lower Priority Power Mode
	OFF_EMU	Transitions to a Lower Priority Power Mode
ON	MEM_RET	Transitions to a Lower Priority Power Mode
	MEM_RET_EMU	Transitions to a Lower Priority Power Mode
	FULL_RET	Transitions to a Lower Priority Power Mode
	MEM_OFF	Transitions to a Lower Priority Power Mode
	FUNC_RET	Transitions to a Lower Priority Power Mode
	WARM_RST	Transitions between ON and WARM_RST
	DBG_RECOV	Transitions to and from DBG_RECOV
WARM_RST	ON	Transitions between ON and WARM_RST
	DBG_RECOV	Transitions to and from DBG_RECOV
DBG_RECOV	ON	Transitions to and from DBG_RECOV

4.2.1 PPU Reset Behavior

The PPU is reset with the **RESETn** input.

The PPU can be configured to enter OFF or ON at reset de-assertion.

Reset to OFF

When the default power mode is OFF, and the reset is de-asserted, the PPU remains in OFF until another policy is programmed in the Power Policy Register (PPU_PWPR) or dynamic transitions are enabled, and a higher priority power mode is requested on the **DEV*ACTIVE** inputs.

It is IMPLEMENTATION DEFINED if there is a transition on the PCSM P-Channel interface. For the PCSM P-Channel interface behavior at reset see *PCSM P-Channel Reset* on page 4-22.

If there is a default transition it does not generate an interrupt event when it completes.

Reset to ON

When the default power mode is ON the PPU needs to control the domain power on and exit from reset. Therefore, there is a transition before the domain reaches ON.

Figure 4-18 shows this sequence for a Q-Channel PPU.

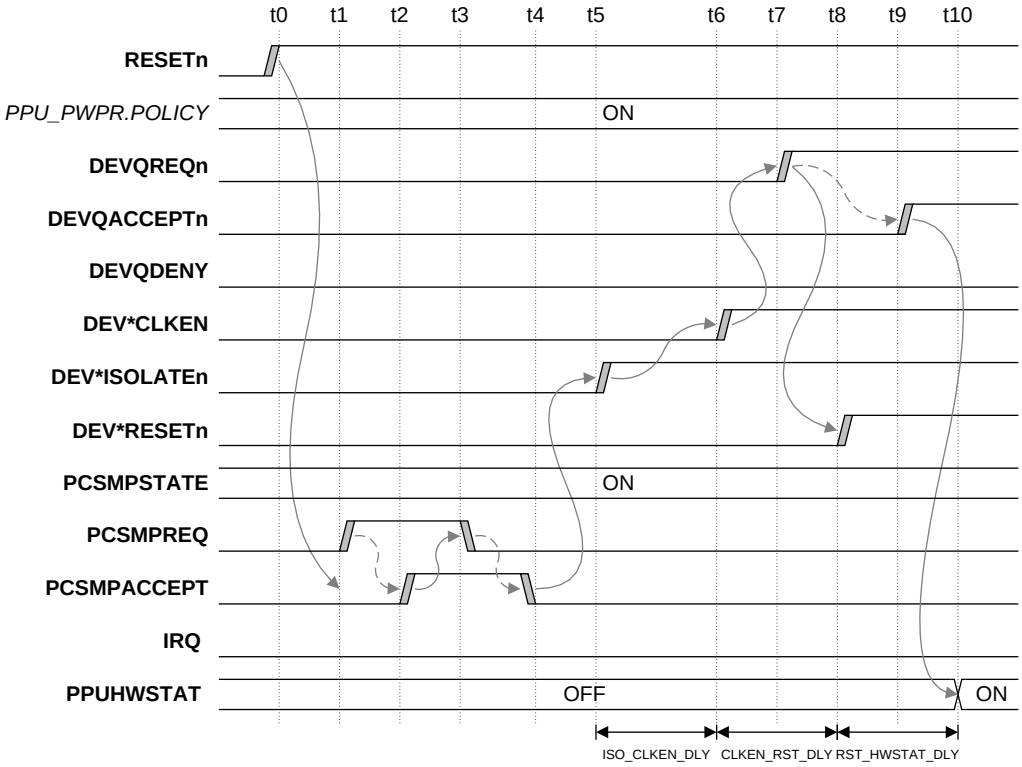

Figure 4-18 Q-Channel PPU reset when default power mode is ON

Figure 4-19 shows this sequence for a P-Channel PPU.

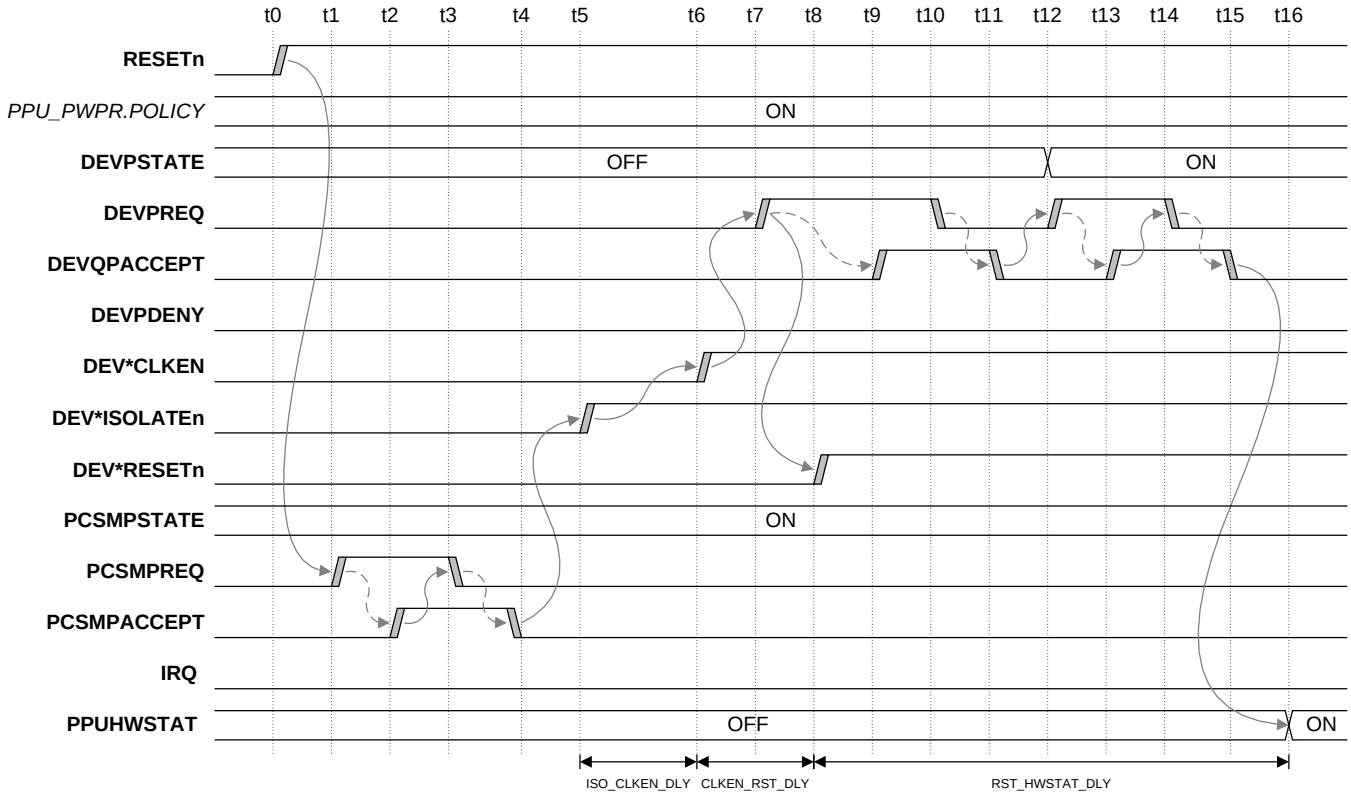

Figure 4-19 P-Channel PPU reset when default power mode is ON

This default transition does not generate an interrupt when completed.

4.2.2 Transitions from OFF or MEM_RET, to a Higher Priority Power Mode

For P-Channel PPU, transitions from OFF or MEM_RET to higher priority power modes, except to DBG_RECOV and MEM_RET, the PPU makes two device interface handshakes. The first handshake initializes the component with the current power mode, and then another handshake moves it to the requested power mode.

Figure 4-20 shows an example of this type of transition for a P-Channel PPU.

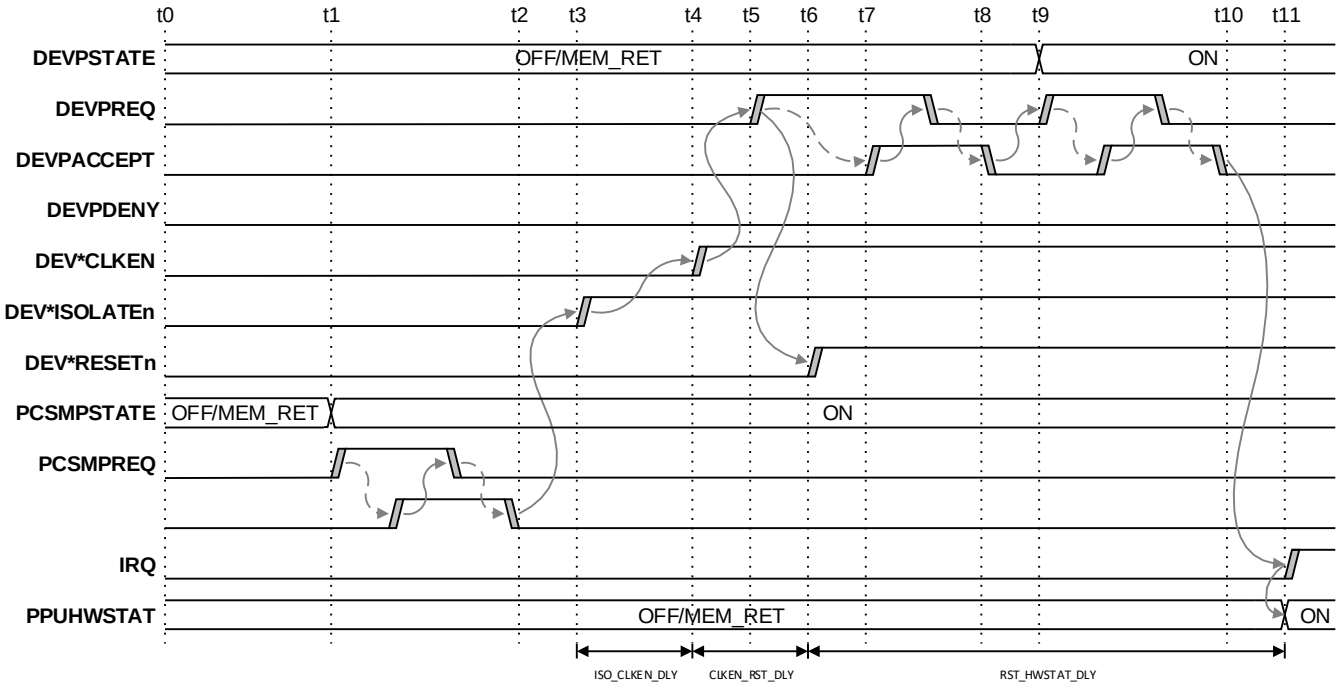

Figure 4-20 Transition from OFF or MEM_RET to ON with a P-Channel PPU

Figure 4-21 shows an example of this type of transition for a Q-Channel PPU.

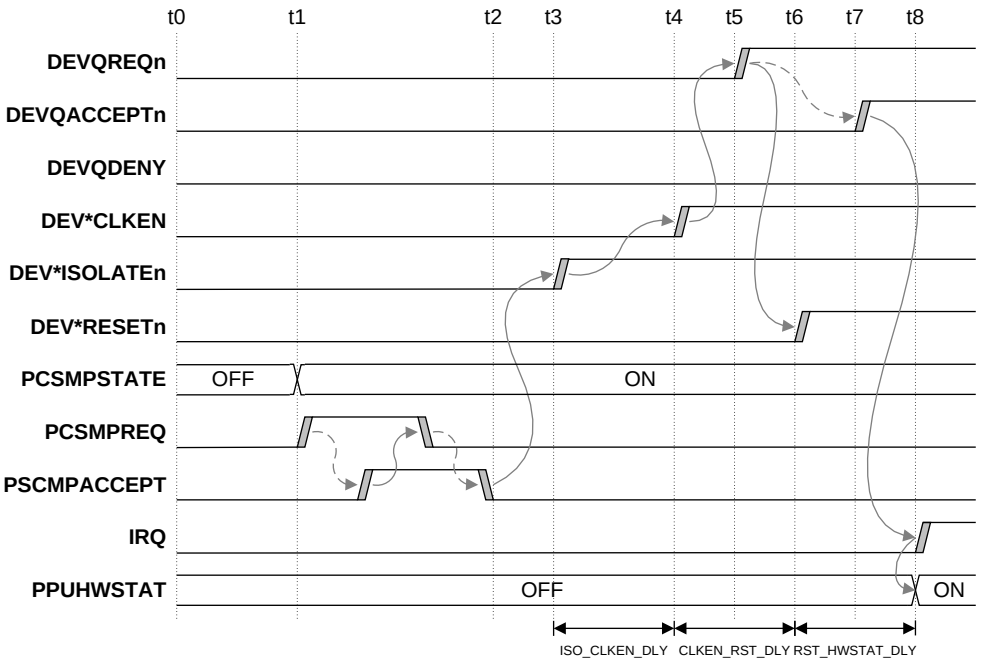

Figure 4-21 Transition from OFF to ON with a Q-Channel PPU

4.2.3 Transitions to a Higher Priority Power Mode

Transitions to a Higher Priority Power Mode with a Q-Channel PPU

Figure 4-22 shows a Q-Channel PPU transition from FULL_RET to ON.

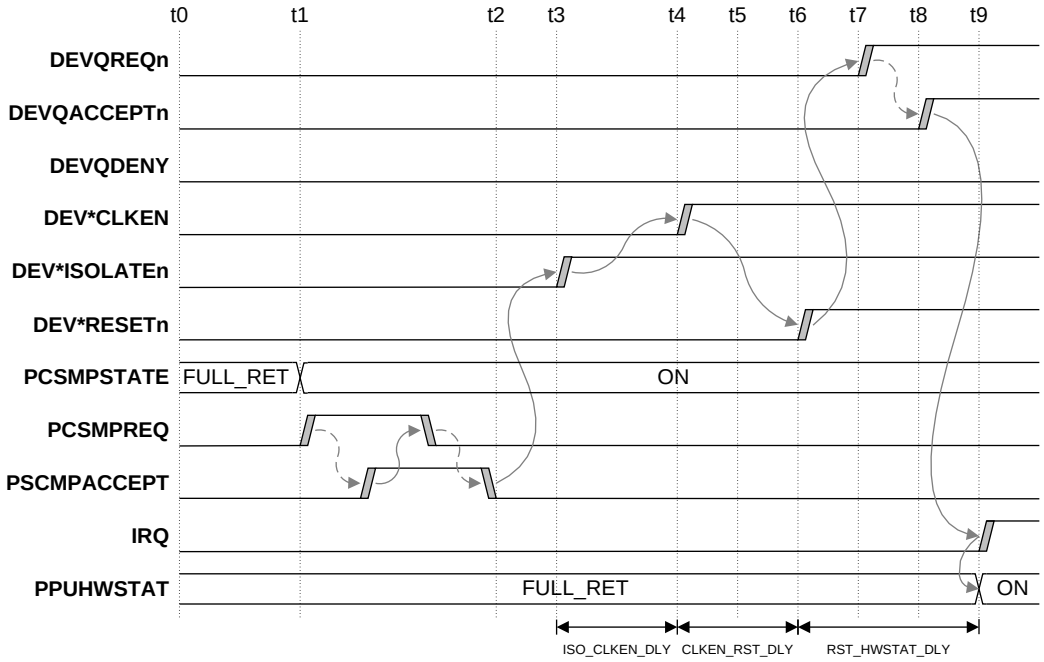

Figure 4-22 Transition from FULL_RET to ON with a Q-Channel PPU

The exact changes to the **DEV*CLKEN**, **DEV*ISOLATEn** and **DEV*RESETn** signals is dependent on the power modes being transitioned between. For details of the device controls application in each mode see *Device Controls* on page 4-16.

Transitions to a Higher Priority Power Mode with a P-Channel PPU

Figure 4-23 shows a P-Channel PPU transition from FULL_RET to ON.

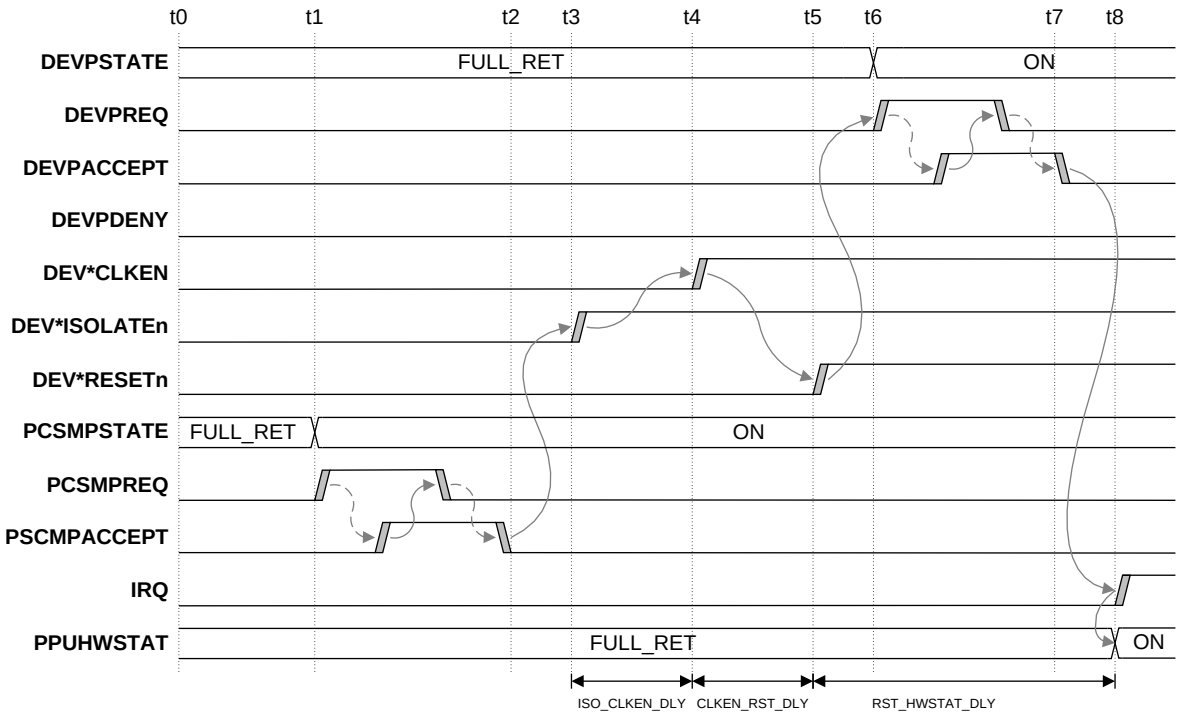

Figure 4-23 Transition from FULL_RET to ON with a P-Channel PPU

The exact changes to the **DEV*CLKEN**, **DEV*ISOLATEn** and **DEV*RESETn** signals is dependent on the power modes being transitioned between. For details of the device controls application in each mode see *Device Controls* on page 4-16.

4.2.4 Transitions to a Lower Priority Power Mode

Transitions to a Lower Priority Power Mode with a Q-Channel PPU

Figure 4-24 shows a Q-Channel PPU transition from ON to FULL_RET.

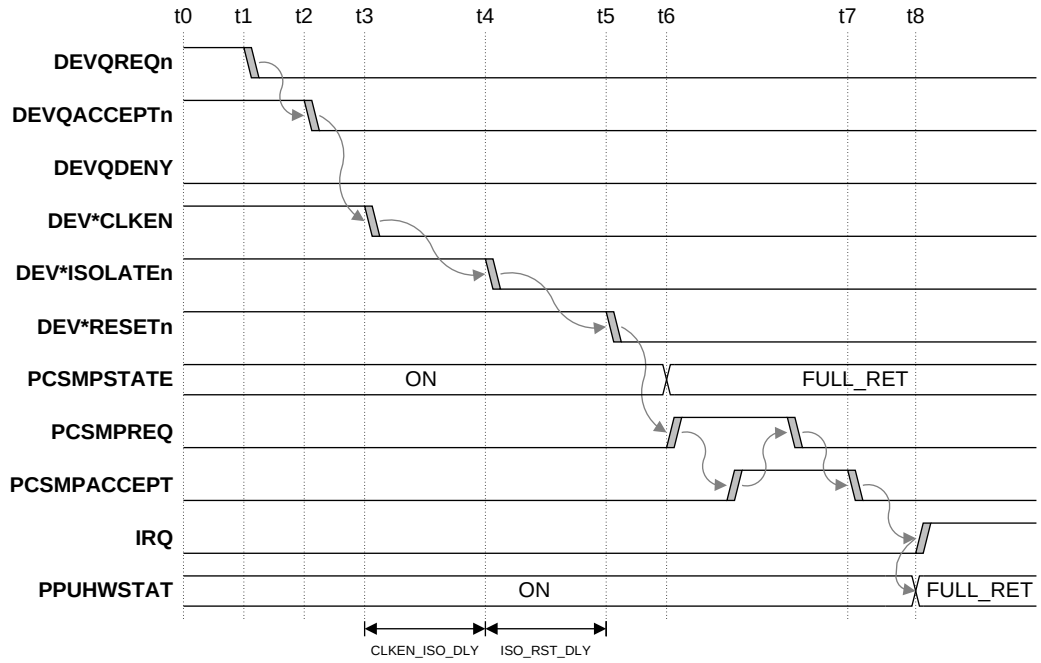

Figure 4-24 Transition from ON to FULL_RET with a Q-Channel PPU

The exact changes to the **DEV*CLKEN**, **DEV*ISOLATEn** and **DEV*RESETn** signals is dependent on the power modes being transitioned between. For details of the device controls application in each mode see *Device Controls* on page 4-16

Transitions to a Lower Priority Power Mode with a P-Channel PPU

Figure 4-25 shows a P-Channel PPU transition from ON to FULL_RET.

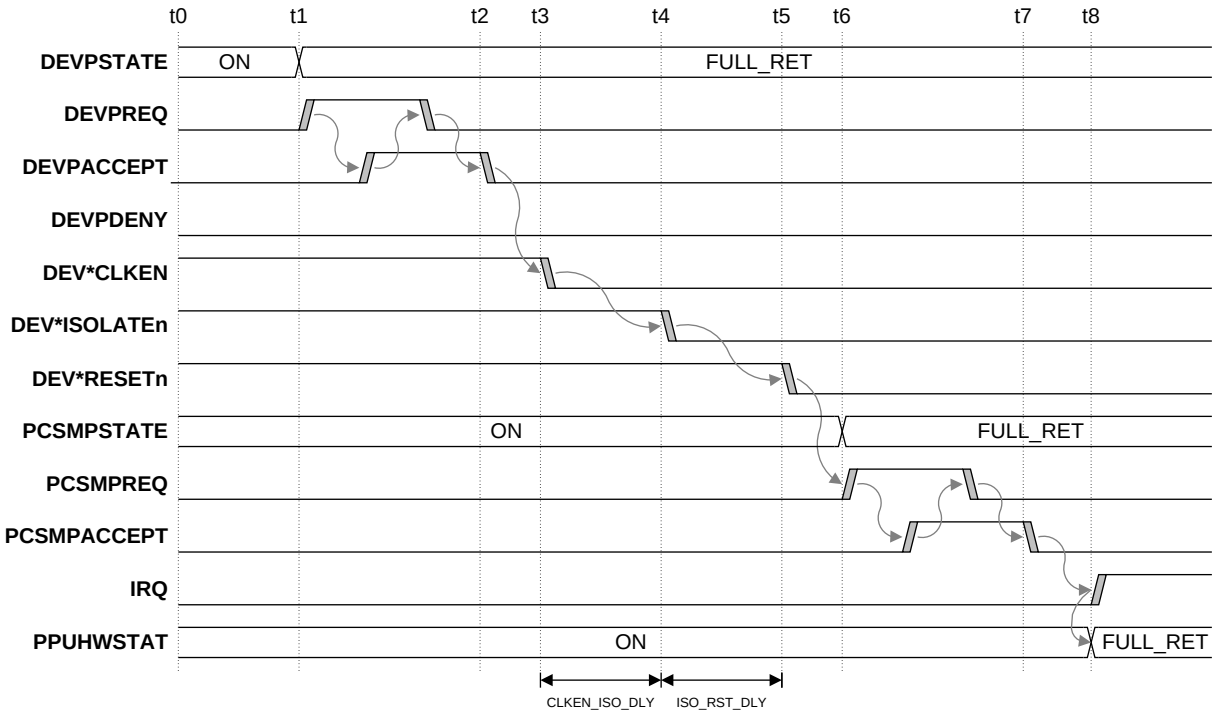

Figure 4-25 Transition from ON to FULL_RET with a P-Channel PPU

The exact changes to the **DEV*CLKEN**, **DEV*ISOLATEn** and **DEV*RESETn** signals is dependent on the power modes being transitioned between. For details of the device controls application in each mode see *Device Controls* on page 4-16.

4.2.5 Transitions between ON and WARM_RST

Figure 4-26 shows a Q-Channel PPU transition from ON to WARM_RST, followed by a transition back to ON when the PPU_PTCR.WARM_RST_DEVREQEN is set to 0b1.

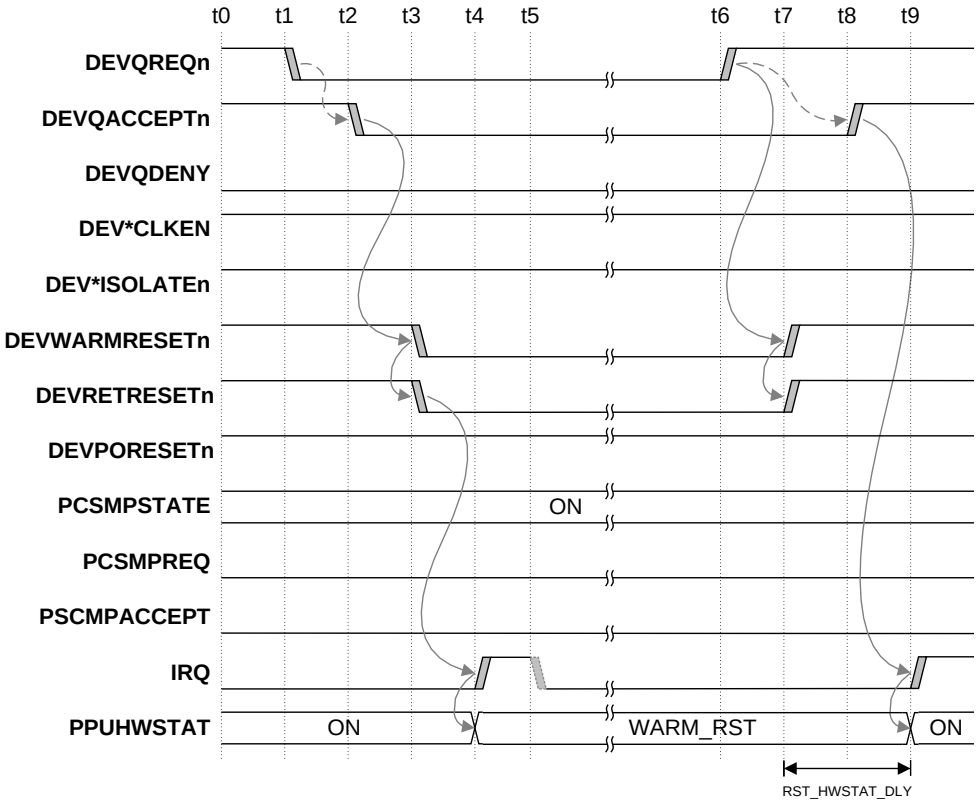

Figure 4-26 Transition from ON to WARM_RST, then back to ON with a Q-Channel PPU

The de-assertion of the IRQ at t5 happens as the result of software clearing the interrupt event and is not required for the transition.

Transitions between ON and WARM_RST for a Q-Channel PPU when the PPU_PTCR.WARM_RST_DEVREQEN is set to 0b0 are the same as in Figure 4-27 but without the device interface handshake that take place from t1 to t2 and from t6 to t8.

Figure 4-27 shows a P-Channel PPU transition from ON to WARM_RST, followed by a transition back to ON when the PPU_PTCR.WARM_RST_DEVREQEN is set to 0b1.

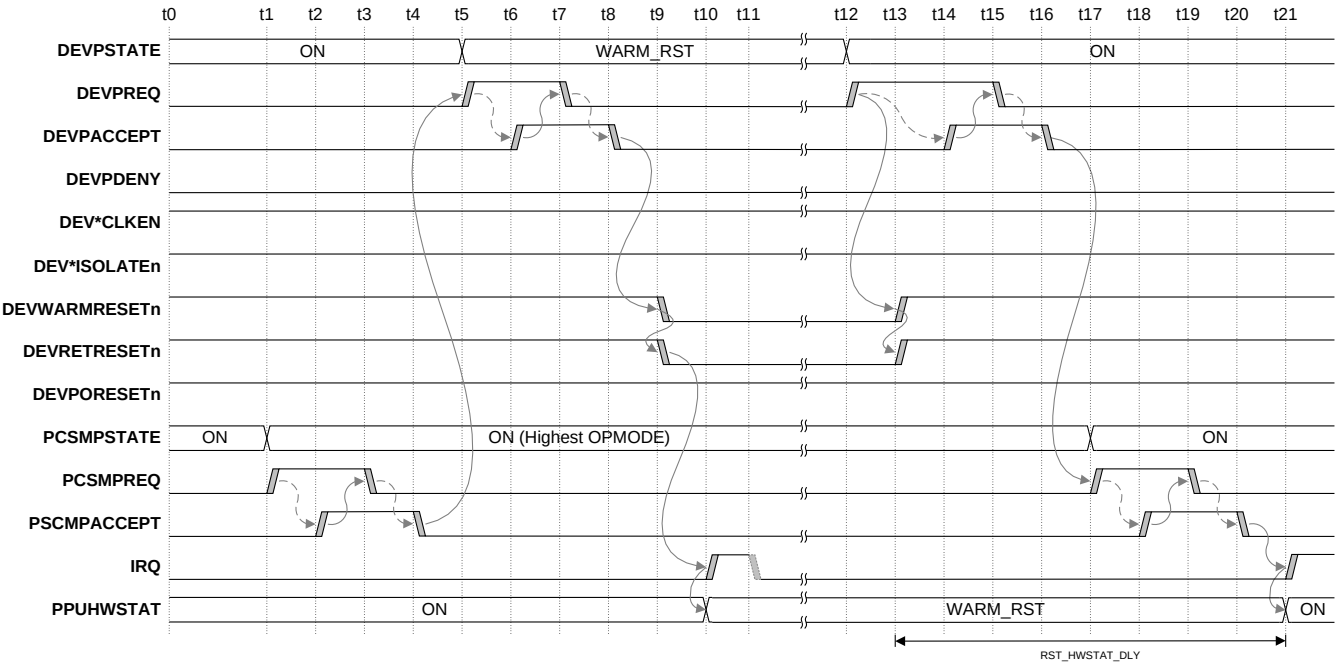

Figure 4-27 Transition from ON to WARM_RST, then back to ON with a P-Channel PPU

The de-assertion of the IRQ at t11 happens as the result of software clearing the interrupt event and is not required for the transition to complete.

When PPU_PTCR.WARM_RST_DEVREQEN is set to 0b0 the device interface P-Channel handshakes do not take place. These device interface P-Channel handshakes are shown in Figure 4-27 between t5 and t8, and between t12 and t16.

Note

The PCSM P-Channel handshake for this transition is not required when either:

- Operating modes are supported but not on the PCSM P-Channel.
    - See *Operating Mode PCSM Transition Configuration* on page 7-7.
- Operating modes are not supported.

This handshake is shown in Figure 4-27 between t1 and t4, and between t17 and t20.

4.2.6 Transitions from OFF to MEM_RET

When the PPU transitions between OFF and MEM_RET it only needs to complete a PCSM handshake. There is no change in conditions of the reset, isolation and clock enable and no device interface handshake as the logic is off in both power modes.

Figure 4-28 shows an example of a transition from OFF to MEM_RET.

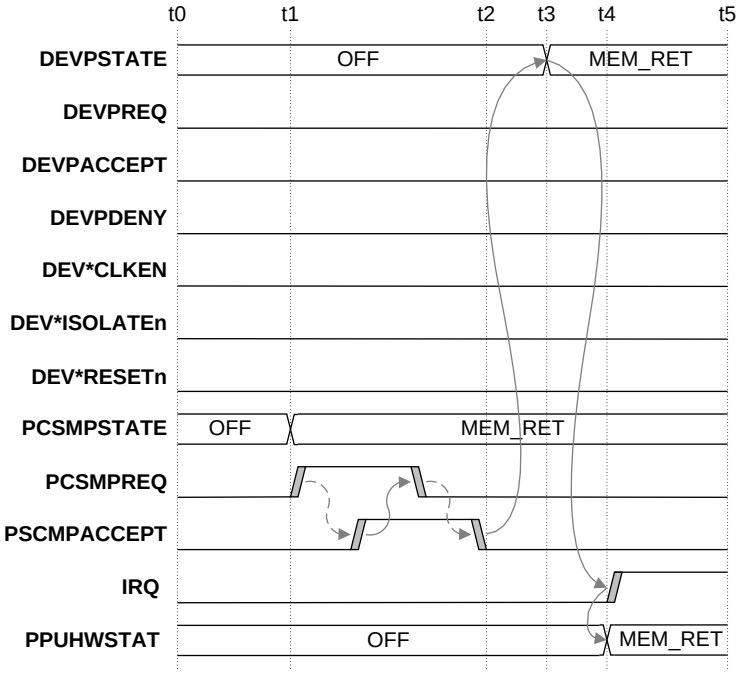

Figure 4-28 Transition from OFF to MEM_RET with a P-Channel PPU

4.2.7 Transitions between OFF_EMU and MEM_RET_EMU

When a P-Channel PPU transitions between OFF_EMU and MEM_RET_EMU it only needs to complete a device interface handshake. There is no change in conditions of the reset, isolation, and clock enable or logic or memory power supply. However, the component needs to know the correct power mode as it might need this to apply internal conditions and for subsequent power mode transitions.

Figure 4-29 shows an example transition from OFF_EMU to MEM_RET_EMU.

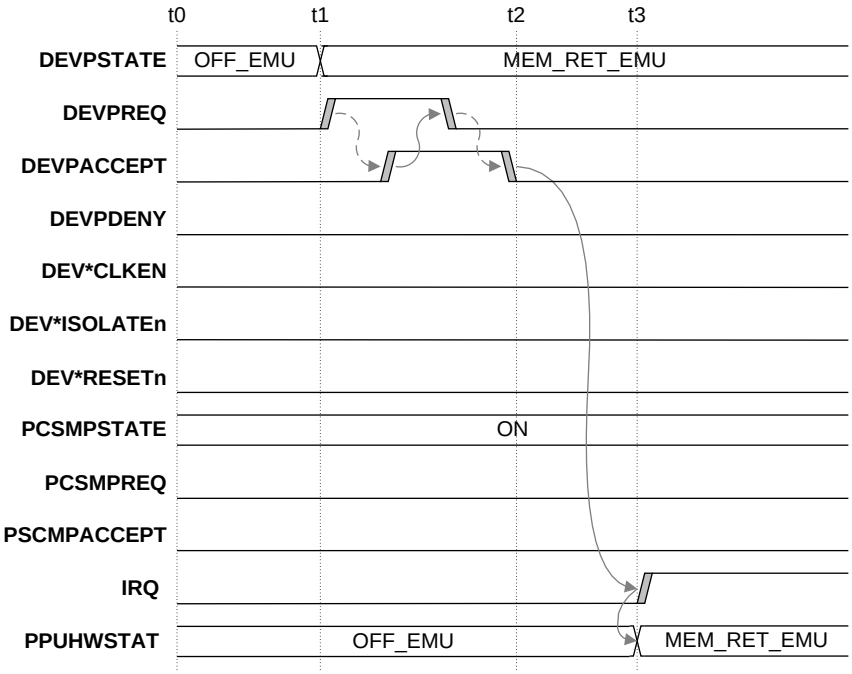

Figure 4-29 Transition from OFF_EMU to MEM_RET_EMU with a P-Channel PPU

When a Q-Channel PPU transitions between OFF_EMU and MEM_RET_EMU it makes no device interface handshake, the PPU only updates **PPUHWSTAT**.

4.2.8 Transitions to and from DBG_RECOV

When a P-Channel PPU transitions to DBG_RECOV, the steps it takes depends on the setting of the PPU_PTCR.DBG_RECOV_PORST_EN bit. The difference sequences are detailed in the following sections.

For more information see *DBG_RECOV and DEVPORESETn* on page 4-18 and *Component Resets* on page 4-19.

Transitions to DBG_RECOV can be made from any other power mode. However, the only transition allowed from DBG_RECOV is to ON.

Transitioning to DBG_RECOV from OFF or MEM_RET, with DBG_RECOV_PORST_EN set to 0b0 is different to entering from all other power modes, see *Transitions to DBG_RECOV asserting DEVWARMRESETn only, from OFF and MEM*_RET on page 4-37.

Transitions to DBG_RECOV asserting DEVWARMRESETn only

Figure 4-30 shows an example entry into DBG_RECOV from FULL_RET when DBG_RECOV_PORST_EN is set to 0b0.

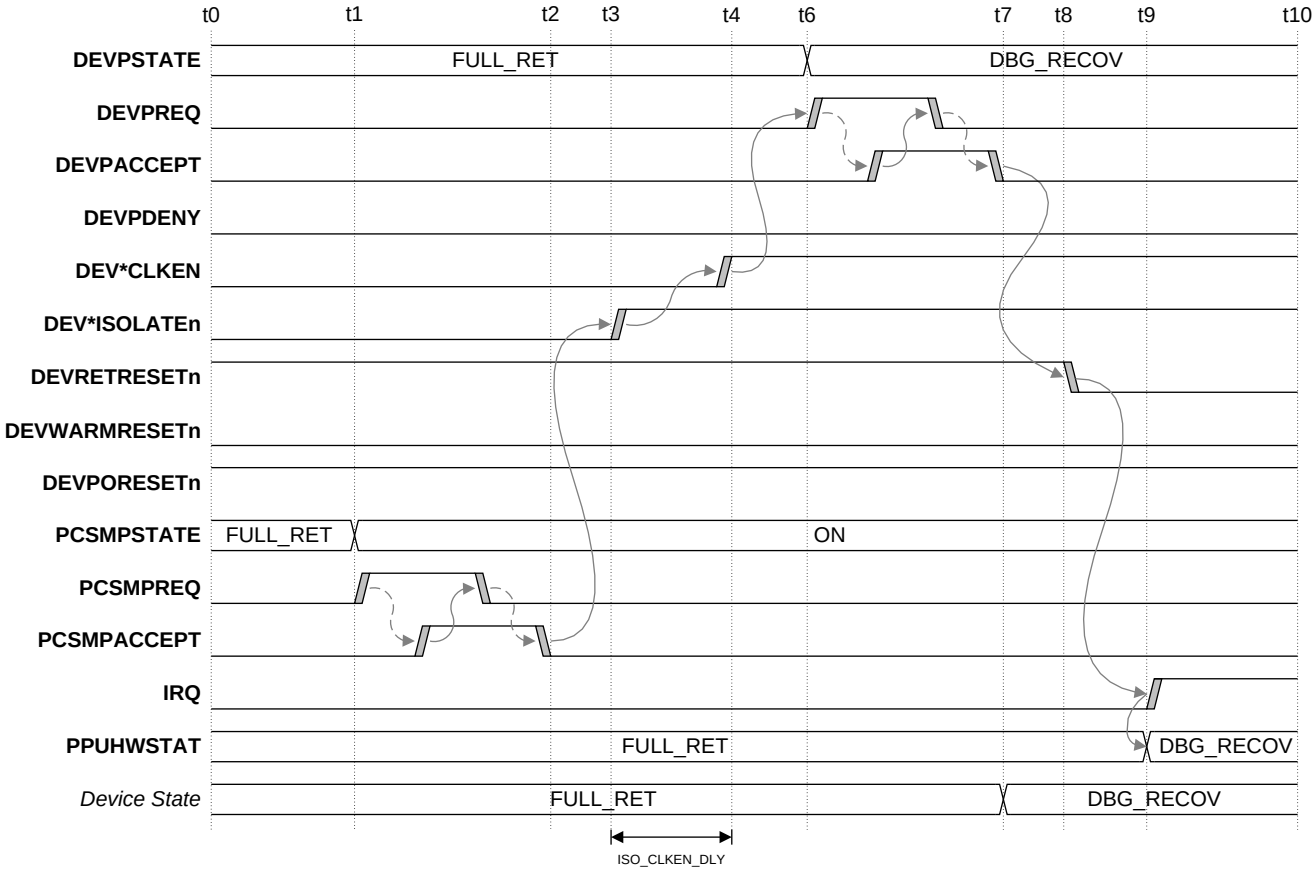

Figure 4-30 FULL_RET to DBG_RECOV, with DBG_RECOV_PORST_EN set to 0b0

The PPU handshakes the PCSM to ON before deactivating any isolation and then enabling clocks, it then performs a device interface handshake to DBG_RECOV. Once this is completed the PPU asserts any required resets.

The exact changes to the **DEV*CLKEN**, **DEV*ISOLATEn,** and **DEV*RESETn** signals is dependent on the power mode that DBG_RECOV is entered from.

For details of the device controls application in each mode see *Device Controls* on page 4-16.

Note

The PCSM P-Channel handshake for this transition is not required when the PPU begins this transition in:

- ON
- WARM_RST
- MEM_RET_EMU
- OFF_EMU.

And either:

- Operating modes are supported but not on the PCSM P-Channel.
    - See *Operating Mode PCSM Transition Configuration* on page 7-7.
- Operating modes are not supported.

This handshake is shown in Figure 4-30 between t1 and t2.

Transitions to DBG_RECOV asserting DEVWARMRESETn only, from OFF and MEM_RET

Figure 4-31 shows an example entry into DBG_RECOV from MEM_RET when DBG_RECOV_PORST_EN is set to 0b0.

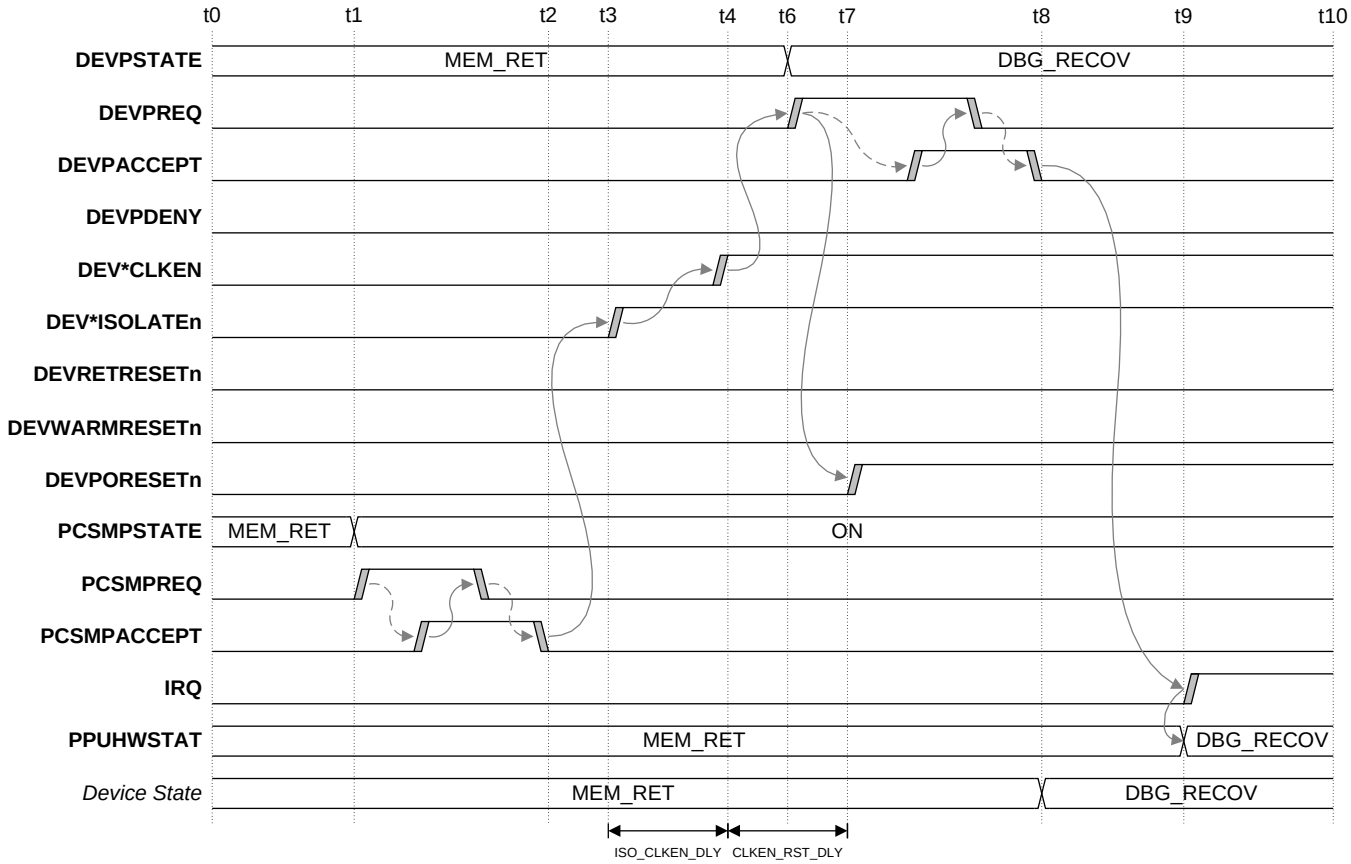

Figure 4-31 MEM_RET to DBG_RECOV, with DBG_RECOV_PORST_EN set to 0b0

Transitions from DBG_RECOV to ON, with DEVWARMRESETn

Figure 4-32 shows a transition from DBG_RECOV to ON when DBG_RECOV_PORST_EN is set to 0b0.

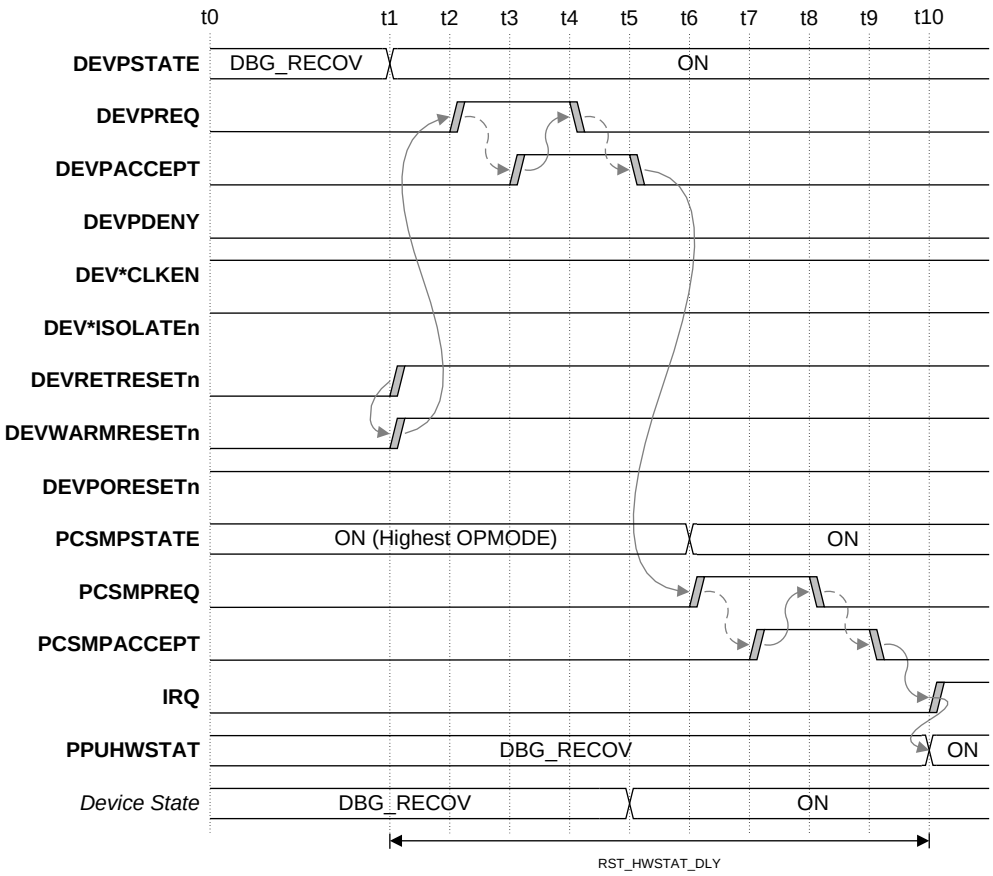

Figure 4-32 Transition from DBG_RECOV to ON, DBG_RECOV_PORST_EN set to 0b0

Note

If operating modes are not supported, the PCSM P-Channel transition, shown in Figure 4-32 between t6 and t9, is not performed, as it only required for an operating mode change.

Transitions to DBG_RECOV asserting DEVPORESETn

Figure 4-33 shows an example entry into DBG_RECOV from FULL_RET when DBG_RECOV_PORST_EN is set to 0b1.

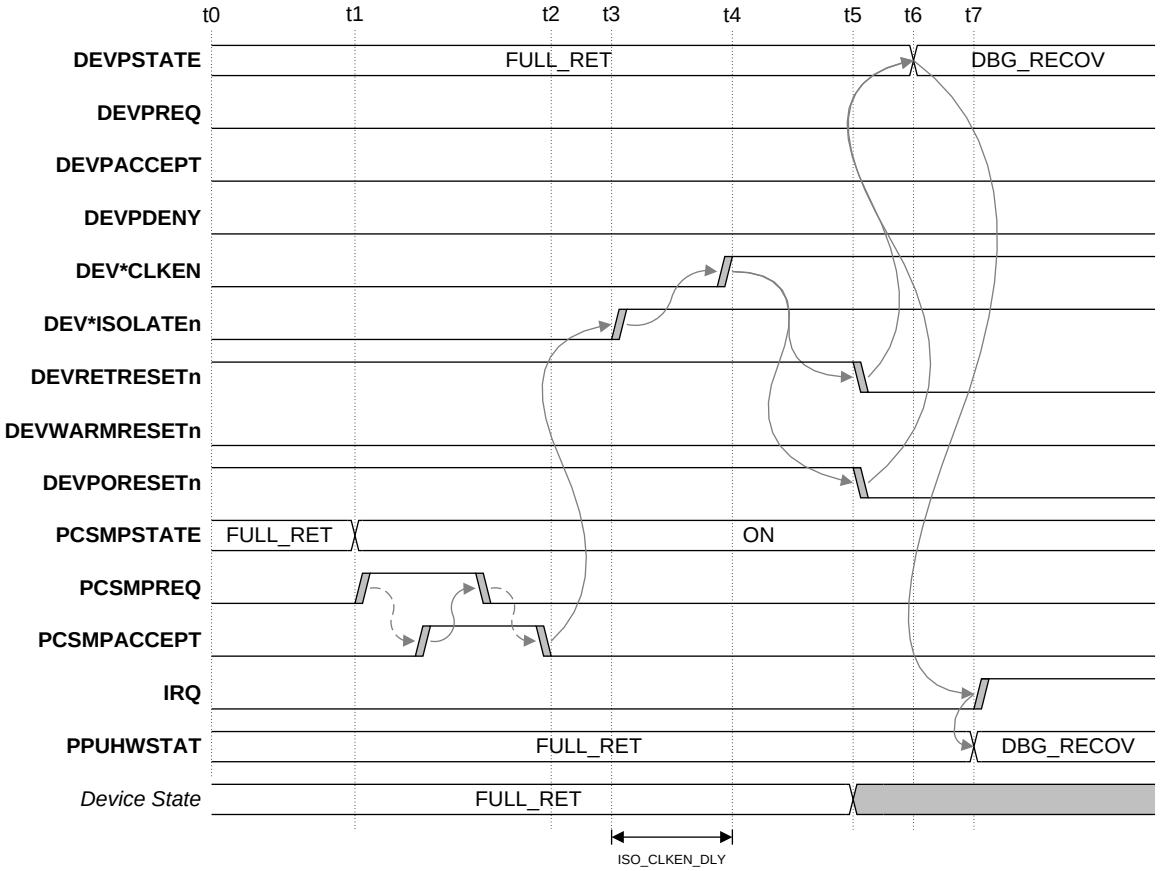

Figure 4-33 Entry into DBG_RECOV from FULL_RET, DBG_RECOV_PORST_EN set to 0b1

The PPU handshakes the PCSM to ON before disabling isolation and enabling clocks, it then asserts any required resets.

The exact changes to the **DEV*CLKEN**, **DEV*ISOLATEn** and **DEV*RESETn** signals is dependent on the power mode that DBG_RECOV is entered from.

For details of the device controls application in each mode see Device Controls on page 4-16.

Note

If the PPU begins this transition in ON, WARM_RST, MEM_RET_EMU, or OFF_EMU, and operating modes are not supported, the PCSM P-Channel transition, shown in Figure 4-33 between t1 and t2, is not required.

Transitions from DBG_RECOV to ON, with DEVPORESETn

Figure 4-34 shows a transition from DBG_RECOV to ON when DBG_RECOV_PORST_EN is set to 0b1.

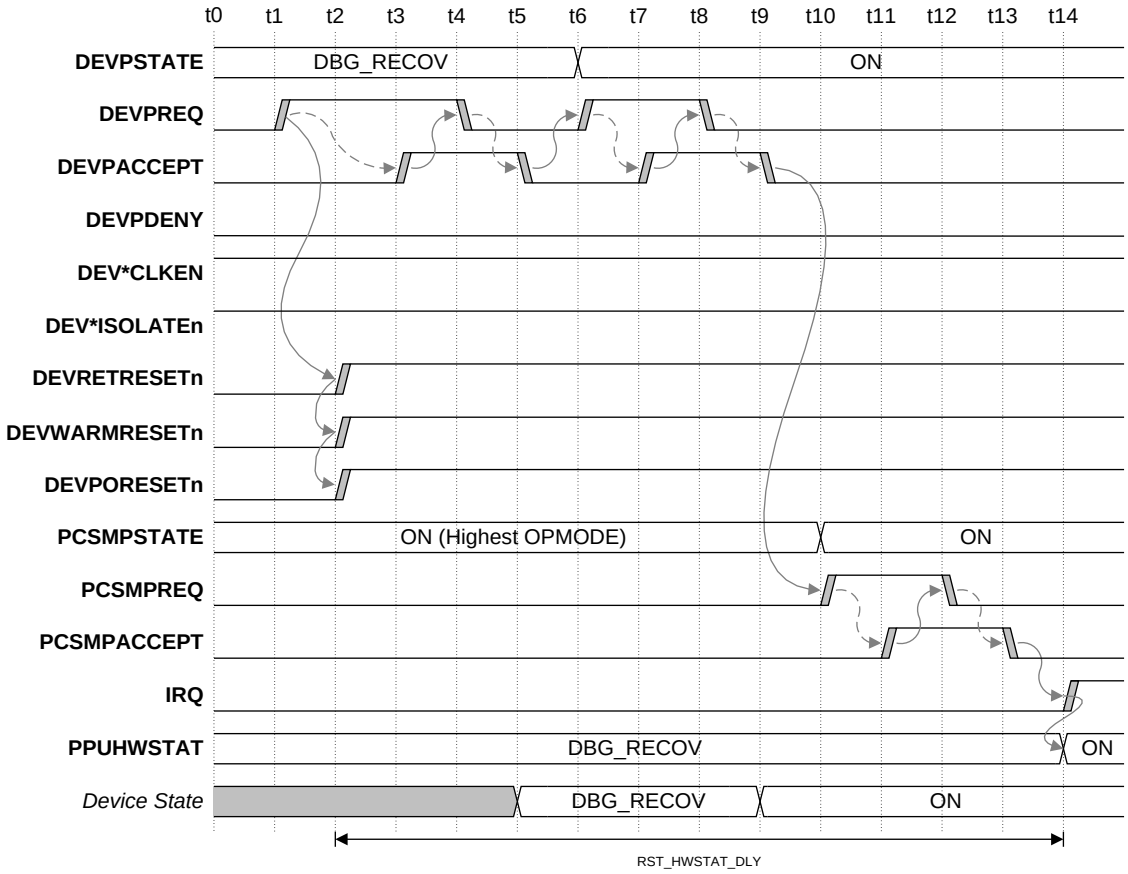

Figure 4-34 Transition from DBG_RECOV to ON, DBG_RECOV_PORST_EN set to 0b1

Note

If operating modes are not supported, the PCSM P-Channel transition, shown in Figure 4-34 between t10 and t13, is not performed, as it only required for an operating mode change.

4.2.9 Transition Denials

If a transition requires the sequencing of PCSM and clock enables, resets, or isolation enables before performing a device interface handshake, if the handshake is denied these actions must be reversed.

This section gives some examples sequences when transitions are denied.

Transition Denial to a Lower Priority Power Mode

Figure 4-35 shows a Q-Channel PPU transition from ON to a lower priority power mode that is denied on the device interface.

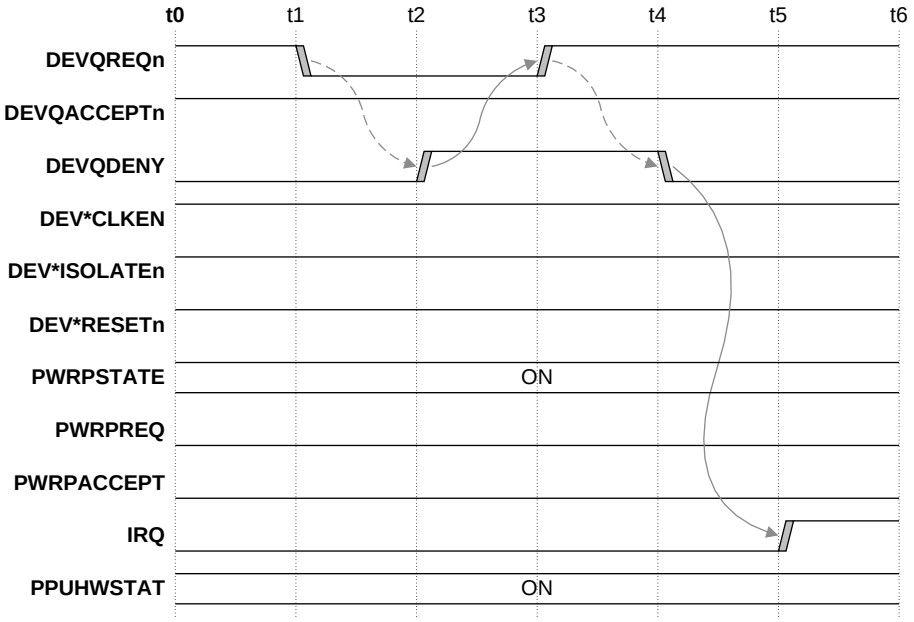

Figure 4-35 Transition denial from ON to a lower priority power mode with a Q-Channel PPU

The sequence is the same for a P-Channel PPU if the Q-Channel denial is replaced with a P-Channel denial in Figure 4-35 between t1 and t4.

Transition Denial to a Higher Priority Power Mode

Figure 4-36 shows a P-Channel PPU transition from FULL_RET to ON being denied.

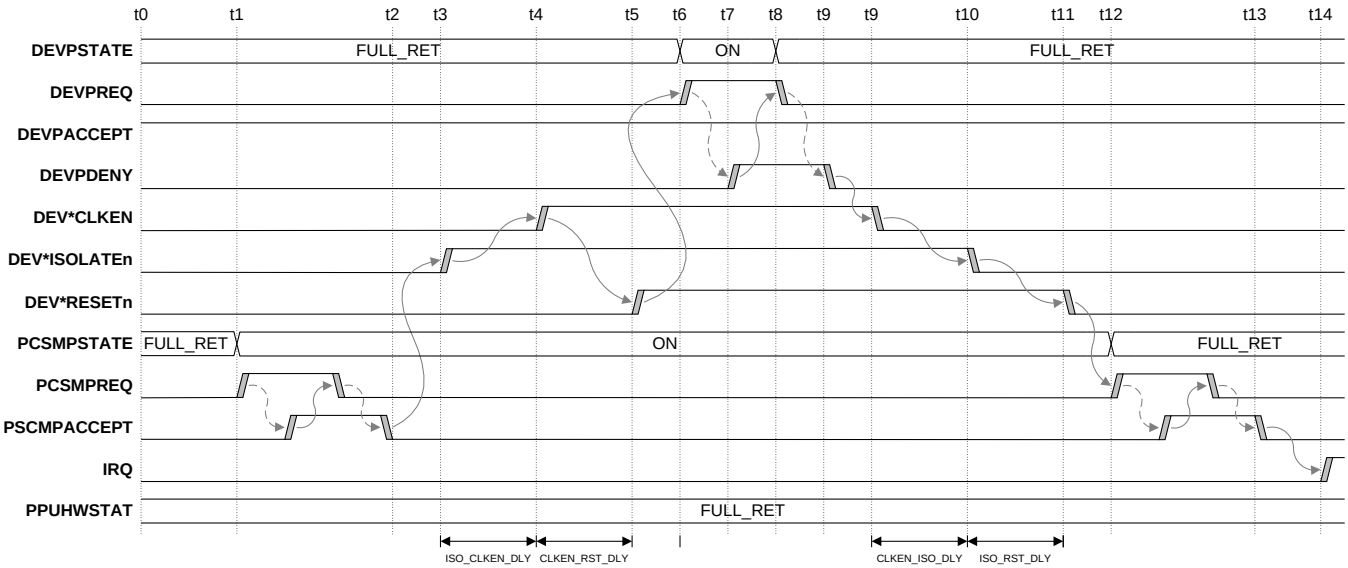

Figure 4-36 P-Channel PPU transition denial from FULL_RET to ON

Note

Device control delays apply when the device controls are reverted following a device interface denial.

Q-Channel PPU transitions to higher priority power modes, except for those to warm reset, cannot be denied.

4.3 Operating Mode Transition Sequences

This section describes the external signaling for all interfaces during operating mode transitions in ON to ON power mode transitions.

Operating mode changes that happen at the same time as power mode transitions follow the power mode transition sequences, for more information see *Power Mode Transition Sequences* on page 4-25.

For some components operating mode transitions do not require modifying any PCSM managed resources therefore these transitions do not require a PCSM P-Channel handshake. A PCSM transition always occurs when there is a power mode change in the same transition.

Therefore, whether the PPU performs PCSM transitions for operating mode transitions can be configured. For more information see *Operating Mode PCSM Transition Configuration* on page 7-7.

The following sections show sequences with PSCM transitions. If the PPU is configured to not perform PCSM transitions for operating mode transitions, then these PCSM transition steps are removed.

4.3.1 Static Use Model

Static use model transitions are the same as those described in *Independent Use Model* on page 4-44.

4.3.2 Ladder Use Model Transitions

When using the ladder use model going to a higher priority operating mode implies an increase in capabilities, the domain keeps all the current capabilities and adds more. Therefore, when moving to a higher priority operating mode the PCSM P-Channel handshake takes place before the device interface handshake to add any capabilities.

Likewise going to a lower priority operating mode implies a decrease in capabilities. Therefore, when moving from to a lower priority operating mode the PCSM P-Channel handshake takes place after the device interface handshake to remove any capabilities.

In these sequences, there are no changes to the **DEV*CLKEN**, **DEV*ISOLATEn**, and **DEV*RESETn** outputs.

Ladder Use Model Transitions to a Higher Priority Operating Mode

Figure 4-37 shows an example ON to ON power mode transition that changes the operating mode from OPMODE_00 to OPMODE_01.

When going to a higher priority operating mode the PCSM P-Channel handshake takes place before the device interface handshake.

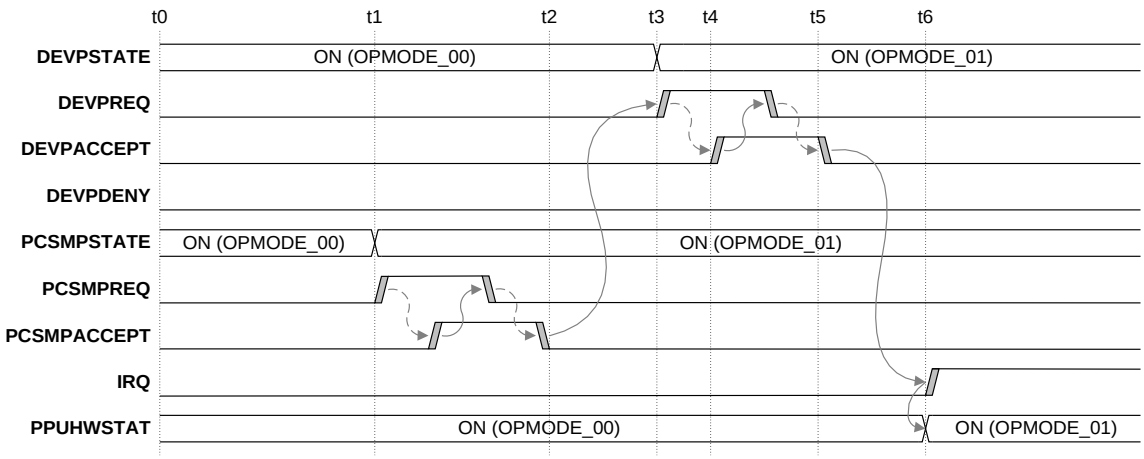

Figure 4-37 Ladder use model transition to higher priority operating mode

Ladder Use Model Transitions to a Lower Priority Operating Mode

Figure 4-38 shows an example ON to ON power mode transition that changes the operating mode from OPMODE_01 to OPMODE_00.

When going to a lower priority operating mode the PCSM P-Channel handshake takes place after the device interface handshake.

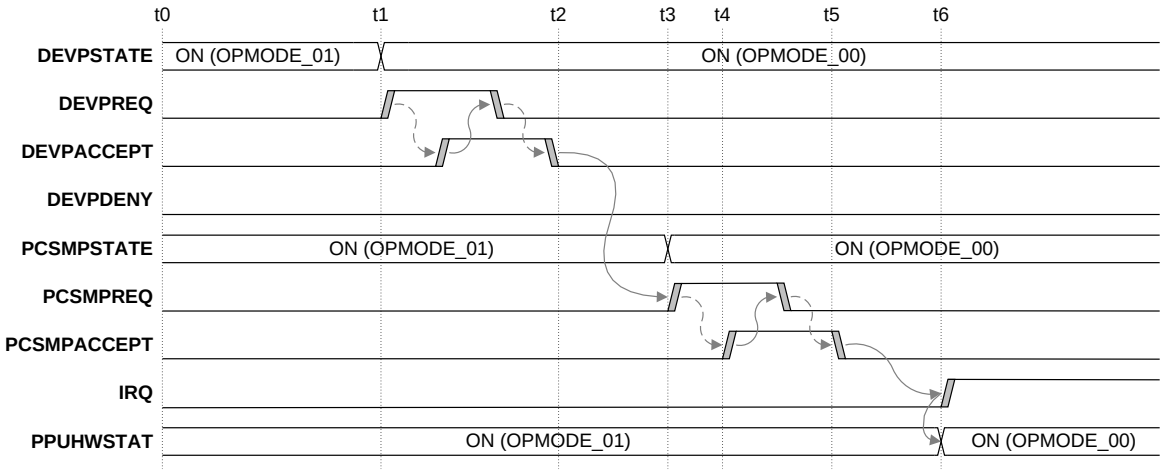

Figure 4-38 Ladder use model transition to lower priority operating mode

4.3.3 Independent Use Model

In the independent use model, the PPU can move directly between operating modes.

If resources are only being added then a PCSM handshake is only required before the device interface handshake, for example a transition from OPMODE_00 to OPMODE_01. An example transition sequence is shown in Figure 4-37 on page 4-43.

If resources are only being removed then a PCSM handshake is only required after the device interface handshake, for example a transition from OPMODE_01 to OPMODE_00. An example transition sequence is shown in Figure 4-38 on page 4-44.

However, an operating mode change can imply both the addition and removal of capabilities, therefore in the general case a PCSM handshake is required both before and after the device interface handshake.

In this example, the component has two resources, in the arrangement in Table 4-15. PCSM support is required to activate and deactivate the resources.

Table 4-15 Operating mode independent DEVPACTIVE mode example

Operating Mode	Resource 0	Resource 1
OPMODE_03	ACTIVE	ACTIVE
OPMODE_02	INACTIVE	ACTIVE
OPMODE_01	ACTIVE	INACTIVE
OPMODE_00	INACTIVE	INACTIVE

Moving from OPMODE_01 to OPMODE_02 requires resource 1 to become active and resource 0 to become inactive.

Resource 1 needs to be made active before the device interface is used to put the component in this mode.

Resource 0 needs to be made inactive after the device interface is used to put the component in this mode.

Therefore, the PCSM needs to be put in OPMODE_03 before the device interface, where both resources are active. Then, after the device interface handshake has been accepted the PCSM is put into OPMODE_02, deactivating resource 0.

Figure 4-39 shows an example of this ON to ON transition that changes the operating mode from OPMODE_01 to OPMODE_02.

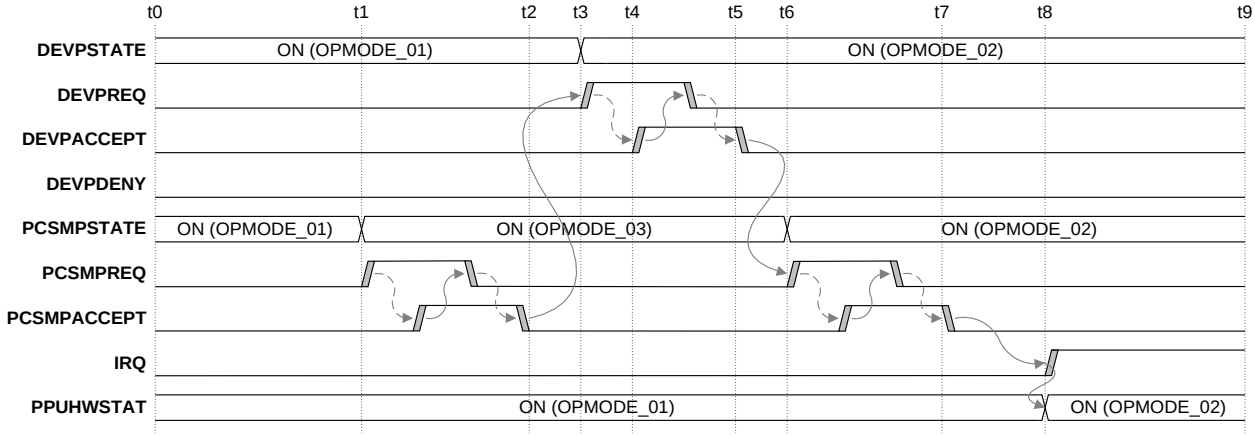

Figure 4-39 Independent use model operating mode transition

If the device interface handshake was denied the PCSM would be put back into OPMODE_01.

5 Programmers Model

This section describes the programmers' model. It contains the following sections:

- *Register Summary* on page 5-2.
- *Register Descriptions* on page 5-3.

5.1 Register Summary

Table 5-1 shows a summary of the PPU registers.

Table 5-1 PPU register summary

Address Offset	Access Type	Name	Short Name	Page
0x000	R/W	Power Policy Register	PPU_PWPR	5-3
0x004	R/W	Power Mode Emulation Register	PPU_PMER	5-4
0x008	RO	Power Status Register	PPU_PWSR	5-4
0x00C	RO	Reserved	-	-
0x010	RO	Device Interface Input Current Status Register	PPU_DISR	5-6
0x014	RO	Miscellaneous Input Current Status Register	PPU_MISR	5-7
0x018	RO	Stored Status Register	PPU_STSR	5-8
0x01C	RW	Unlock register	PPU_UNLK	5-8
0x020	R/W	Power Configuration Register	PPU_PWCR	5-9
0x024	R/W	Power Mode Transition Configuration Register	PPU_PTCR	5-10
0x030	R/W	Interrupt Mask Register	PPU_IMR	5-12
0x034	R/W	Additional Interrupt Mask Register	PPU_AIMR	5-13
0x038	R/W	Interrupt Status Register	PPU_ISR	5-13
0x03C	R/W	Additional Interrupt Status Register	PPU_AISR	5-15
0x040	R/W	Input Edge Sensitivity Register	PPU_IESR	5-15
0x044	R/W	Operating Mode Active Edge Sensitivity Register	PPU_OPSR	5-17
0x050	R/W	Functional Retention RAM Configuration Register	PPU_FUNRR	5-18
0x054	R/W	Full Retention RAM Configuration Register	PPU_FULRR	5-18
0x058	R/W	Memory Retention RAM Configuration Register	PPU_MEMRR	5-19
0x160	R/W	Power Mode Entry Delay Register 0	PPU_EDTR0	5-20
0x164	R/W	Power Mode Entry Delay Register 1	PPU_EDTR1	5-20
0x170	R/W	Device Control Delay Configuration Register 0	PPU_DCDR0	5-21
0x174	R/W	Device Control Delay Configuration Register 1	PPU_DCDR1	5-22
0xFB0	RO	PPU Identification Register 0	PPU_IDR0	5-22
0xFB4	RO	PPU Identification Register 1	PPU_IDR1	5-24
0xFC8	RO	Implementation Identification Register	PPU_IIDR	5-26
0xFCC	RO	Architecture Identification Register	PPU_AIDR	5-26
0xFD0-FFF	RO	IMPLEMENTATION DEFINED Identification Registers	-	-

5.2 Register Descriptions

All reserved bits are Write Ignored and Read as Zero (WI/RAZ).

5.2.1 Power Policy Register (PPU_PWPR)

This register enables software to program both power and operating mode policy. It also contains related settings including the enable for dynamic transitions and the lock enable.

This register does not reflect the current power mode value. The current power mode of the domain is reflected in the Power Status Register (PPU_PWSR).

Table 5-2 Power Policy Register (PPU_PWPR)

Bits	Type	Default	Name	Function
[31:25]	RO	0x0	Reserved	Reserved
[24]	RW	DEF_OP_DYN_EN	OP_DYN_EN	Operating mode dynamic transition enable. When this bit is set to 0b1 dynamic transitions are enabled for operating modes, allowing transitions to be initiated by changes on operating mode **DEVACTIVE** inputs. For further information see *Operating Mode Transition Rules* on page 3-24. When operating modes are not supported this bit is reserved.
[23:20]	RO	0x0	Reserved	Reserved
[19:16]	RW	DEF_OP_POLICY	OP_POLICY	Operating mode policy. When static operating mode transitions are enabled, OP_DYN_EN is set to 0b0, then this is the target operating mode for the PPU. When dynamic operating mode transitions are enabled, OP_DYN_EN is set to 0b1, then this is the minimum operating mode for the PPU. Operating mode policy enumerations are described in *Operating Mode Values* on page 3-7. The default value for this field depends upon DEF_OP_POLICY and the PCSMMODESTAT input. For further information see *Default Operating Policy* on page 7-6 and PCSMMODESTAT on page 4-24.. When operating modes are not supported this field is reserved.
[15:13]	RO	0x0	Reserved	Reserved
[12]	RW	0x0	LOCK_EN	Lock enable bit. For further information see *Lock* on page 3-14. This bit can be set regardless of the PWR_DYN_EN bit setting. If the lock function is not supported this bit is reserved.
[11:9]	RO	0x0	Reserved	Reserved
[8]	RW	DEF_PWR_DYN_EN	PWR_DYN_EN	Power mode dynamic transition enable. When this bit is set to 0b1 dynamic transitions are enabled for power modes, allowing transitions to be initiated by changes on power mode **DEVACTIVE** inputs. For further information see *Dynamic Power Mode Transitions* on page 3-13. If dynamic modes are not supported this bit is reserved.
[7:4]	RO	0x0	Reserved	Reserved

[3:0]	RW	DEF_PWR_POLICY	PWR_POLICY	Power mode policy. When static power mode transitions are enabled, PWR_DYN_EN is set to 0b0, this is the target power mode for the PPU. When dynamic power mode transitions are enabled, PWR_DYN_EN is set to 0b1, this is the minimum power mode for the PPU. Power mode policy enumerations are described in *PPU Mode Values* on page 3-6. The default value for this field depends upon DEF_PWR_POLICY and the PCSMMODESTAT input. For further information see *Configuration Options* on page 7-1 and PCSMMODESTAT on page 4-24.

5.2.2 Power Mode Emulation Enable Register (PPU_PMER)

This register allows software to enable entry into emulated modes. The use of this register is described in *Power Mode Emulation Programming* on page 4-4.

Note

When neither MEM_RET_EMU nor OFF_EMU is supported this register is reserved.

Table 5-3 Power Mode Emulation Enable Register (PPU_PMER)

Bits	Type	Default	Name	Function
[31:1]	RO	0x0	Reserved	Reserved
[0]	RW	0x0	EMU_EN	Power mode emulation enable. When this bit is set to 0b1 transitions to OFF and MEM_RET instead transition to OFF_EMU and MEM_RET_EMU. The use of this bit is described in *Power Mode Emulation Programming* on page 4-4. When neither MEM_RET_EMU nor OFF_EMU is supported this bit is reserved.

5.2.3 Power Status Register (PPU_PWSR)

This read-only register contains status information for the power mode, operating mode, dynamic transitions, and lock feature.

Table 5-4 Power Status Register (PPU_PWSR)

Bits	Type	Default	Name	Function
[31:13]	RO	0x0	Reserved	Reserved.
[24]	RO	DEF_OP_DYN_EN	OP_DYN_STATUS	Operating mode dynamic transition status. When set to 0b1 operating mode dynamic transitions are enabled. There might be a delay in dynamic transitions becoming active or inactive if the PPU is transitioning when OP_DYN_EN is programmed. If operating modes are not supported this bit is reserved.
[23:20]	RO	0x0	Reserved	Reserved.
[19:16]	RO	PCSMMODESTAT[7:4] [a]	OP_STATUS	Operating mode status. These bits reflect the current operating mode of the PPU. Operating mode policy enumerations are described in *Operating Mode Values* on page 3-7. When operating modes are not supported this field is reserved. In the OFF, OFF_EMU, DBG_RECOV, and WARM_RST power modes, this field reflects the current programmed OP_POLICY even though the operating mode **DEVPSTATE** output bits are set to zero.
[15:13]	RO	0x0	Reserved	Reserved.
[12]	RO	0x0	LOCK_STATUS	Lock status. See *Lock* on page 3-14. When set to 0b1 the PPU is locked in the current mode. When set to 0b0 the PPU is not locked in the current mode. If lock is not supported this bit is reserved.
[11:9]	RO	0x0	Reserved	Reserved.
[8]	RO	DEF_PWR_DYN_EN	PWR_DYN_STATUS	Power mode dynamic transition status. When set to 0b1 power mode dynamic transitions are enabled. There might be a delay in dynamic transitions becoming active or inactive if the PPU is transitioning when PWR_DYN_EN is programmed.
[7:4]	RO	0x0	Reserved	Reserved.
[3:0]	RO	PCSMMODESTAT[3:0] [a]	PWR_STATUS	Power mode status. These bits reflect the current power mode of the PPU. Power mode enumerations are described in *PPU Mode Values* on page 3-6.

[a] This value is set dependent on the value of **PCSMMODESTAT** input at reset exit. Only **PCSMMODESTAT[3:0]** values of 0x0000 and 0x0010 are supported. If the **PCSMMODESTAT** signal is not present the default value is 0x0. For further information see *PCSMMODESTAT* on page 4-24. These reset values are regardless of the default policy value, DEF_PWR_POLICY. If the default policy is ON, the PPU transitions to ON from the power mode specified by PCSMMODESTAT and PWR_STATUS updates when this transition is complete.

5.2.4 Device Interface Input Current Status Register (PPU_DISR)

This read-only register contains status reflecting the values of the device interface inputs.

Note

For P-Channel PPU, the PWR_DEVACTIVE_STATUS field LSB representing **DEVPACTIVE[0]** is reserved. This bit represents OFF mode and is not useful for indicating component activity.

Any unused bits are the MSB's and are reserved.

Table 5-5 Device Interface Current Status Register (PPU_DISR)

Bits	Type	Default	Name	Function
[31:24]	RO	0x0	OP_DEVACTIVE_STATUS	Status of the operating mode **DEVACTIVE** inputs. When operating modes are not supported this field is reserved.
[23:11]	RO	0x0	Reserved	Reserved
[10:0]	RO	0x0	PWR_DEVACTIVE_STATUS	Status of the power mode **DEVACTIVE** inputs. *For Q-Channel*: There is one bit for each device interface Q-Channel **DEVQACTIVE**. For example, bit 0 is for the Q-Channel 0 **DEVQACTIVE**, and bit 1 for the Q-Channel 1 **DEVQACTIVE**. For a Q-Channel PPU bits [10:8] are always reserved as only 8 device interface Q-Channels are supported. *For P-Channel*: There is one bit for each bit of the power mode **DEVPACTIVE** inputs. For example, bit 10 is for **DEVPACTIVE**[10], and bit 9 is for **DEVPACTIVE**[9]. Bit 0 relates to **DEVPACTIVE**[0] and is always 0.

5.2.5 Miscellaneous Input Current Status Register (PPU_MISR)

This read-only register contains status reflecting the values of miscellaneous inputs.

Table 5-6 Miscellaneous Input Current Status Register (PPU_MISR)

Bits	Type	Default	Name	Function
[31:24]	RO	0x0	Reserved	Reserved.
[23:16]	RO	0x0	DEVDENY_STATUS	Status of the device interface **DEVDENY** inputs. *For Q-Channel*: There is one bit for each device interface **DEVQDENY**. For example, bit 16 is for Q-Channel 0 **DEVQDENY**, and bit 17 for Q-Channel 1 **DEVQDENY**. *For P-Channel*: Bit 16 is for the device interface **DEVPDENY**. Other bits are reserved.
[15:8]	RO	0x0	DEVACCEPT_STATUS	Status of the device interface **DEVACCEPT** inputs. *For Q-Channel*: There is one bit for each device interface **DEVQACCEPTn**. For example, bit 8 is for Q-Channel 0 **DEVQACCEPTn** and bit 9 for Q-Channel 1 **DEVQACCEPTn**. *For P-Channel*: Bit 8 is for the device interface **DEVPACCEPT**. Other bits are reserved.
[7:1]	RO	0x0	Reserved	Reserved.
[0]	RO	0x0	PCSMPACCEPT_STATUS	The status of the **PCSMPACCEPT** input.

5.2.6 Stored Status Register (PPU_STSR)

This read-only register contains status information about the **DEVDENY** inputs that were set HIGH during the last transition.

This register updates when a transition completes. Therefore, it contains the status of the last transition. This allows software to determine the components that have denied a device interface request after the transition.

Note

For a P-Channel PPU or a Q-Channel PPU with a single Q-Channel this register is reserved.

Table 5-7 Stored Status Register (PPU_STSR)

Bits	Type	Default	Name	Function
[31:8]	RO	0x0	Reserved	Reserved.
[7:0]	RO	0x0	STORED_DEVDENY	Status of the **DEVDENY** signals from the last device interface Q-Channel transition. *For Q-Channel:* There is one bit for each device interface **DEVQDENY**. For example, bit 0 is for Q-Channel 0 **DEVQDENY**, and bit 1 for Q-Channel 1 **DEVQDENY**. For a Q-Channel PPU with a single Q-Channel this field is reserved. *For P-Channel*: This field is reserved.

5.2.7 Unlock Register (PPU_UNLK)

This write-only register allows software to unlock the PPU from a locked power mode.

For more information on the use of this register see *Lock* on page 3-14.

Note

Lock functionality is an optional feature. Whether it is supported can be determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24. When it is not supported, this register is reserved.

Table 5-8 Unlock Register (PPU_UNLK)

Bits	Type	Default	Name	Function
[31:1]	RO	0x0	Reserved	Reserved.
[0]	WO	0x0	UNLOCK	When 0b1 is written to this bit the PPU is unlocked from a locked power mode. A read always returns 0b0. See *Lock* on page 3-14. When lock is not supported this field is reserved.

5.2.8 Power Configuration Register (PPU_PWCR)

This register controls enabling and disabling of hardware control inputs to the PPU.

Note

Before software programs the DEVREQEN bits it must configure the PPU for static transitions and ensure the requested power mode has been reached, this means that no further transitions can occur, otherwise behavior is UNPREDICTABLE.

Additionally, when using a Q-Channel PPU the power mode must be ON before software enables any previously disabled Q-Channels otherwise behavior is UNPREDICTABLE.

The PWR_DEVACTIVEEN and OP_DEVACTIVEEN fields in this register control the ability of the **DEVACTIVE** inputs to initiate power mode transitions, but not the ability to generate input edge interrupt events.

Bits for unused channels are the MSB's and are reserved.

Table 5-9 Power Configuration Register (PPU_PWCR)

Bits	Type	Default	Name	Description
[31:24]	RW	See Table 5-11	OP_DEVACTIVEEN	These bits enable the operating mode **DEVACTIVE** inputs. When a bit is to 0b1 the related **DEVACTIVE** input is enabled, when set to 0b0 it is disabled. All supported bits are reset to 0b1. When operating modes are not supported this field is reserved. For more information see *DEVACTIVE Enables* on page 4-13.
[23:19]	RO	0x0	Reserved	Reserved.
[18:8]	RW	See Table 5-10	PWR_DEVACTIVEEN	These bits enable the power mode **DEVACTIVE** inputs. When a bit is to 0b1 the related **DEVACTIVE** input is enabled, when set to 0b0 it is disabled. All available bits are reset to 0b1. *For Q-Channel*: There is one bit for each device interface Q-Channel **DEVQACTIVE**. For example, bit 8 is for the Q-Channel 0 **DEVQACTIVE**, and bit 9 for the Q-Channel 1 **DEVQACTIVE**. For a Q-Channel PPU bits [18:16] are always reserved as only 8 device interface Q-Channels are supported. *For P-Channel*: There is one bit for each bit of the **DEVPACTIVE** input. For example, bit 18 is for **DEVPACTIVE**[10], and bit 17 is for **DEVPACTIVE**[9]. Bit 8 relates to **DEVPACTIVE**[0] and is always 0b0. For more information see *DEVACTIVE Enables* on page 4-13.
[7:0]	RW	See Table 5-10	DEVREQEN	When set to 0b1 enables the device interface handshake for transitions. All available bits are reset to 0b1. *For Q-Channel*: There is one bit for each device interface channel. For example, bit 0 is for Q-Channel 0, and bit 1 is for Q-Channel 1. *For P-Channel:* Bit 0 is for the single P-Channel. Other bits are reserved. For more information see *Device Interface Channel Enables* on page 4-13.

The default value for the PWR_DEVACTIVEEN and DEVREQEN fields depends upon the PPU configuration. Table 5-10 shows the possible default values.

Table 5-10 PWR_DEVACTIVEEN and DEVREQEN default values

PPU configuration	PWR_DEVACTIVEEN default value	DEVREQEN default value
8 Q-Channels	0x0FF	0xFF
7 Q-Channels	0x07F	0x7F
6 Q-Channels	0x03F	0x3F
5 Q-Channels	0x01F	0x1F
4 Q-Channels	0x00F	0x0F
3 Q-Channels	0x007	0x07
2 Q-Channels	0x003	0x03
1 Q-Channel	0x001	0x01
P-Channel	0x7FE	0x01

The default value for the OP_DEVACTIVEEN field depends upon the PPU configuration. Table 5-11 shows the possible default values.

Table 5-11 OP_DEVACTIVEEN default values

Operating mode use model	NUM_OPMODE_CFG	OP_DEVACTIVEEN default value
Ladder	0	0x00
	1	0x01
	2	0x03
	3	0x07
	4	0x0F
	5	0x1F
	6	0x3F
	7	0x7F
	8	0xFF
Independent	0	0x00
	1	0x01
	3	0x03
	7	0x07
	15	0x0F

5.2.9 Power Mode Transition Configuration Register (PPU_PTCR)

This register contains settings which affect the behaviour of certain power mode transitions.

Table 5-12 Power Mode Transition Configuration Register (PPU_PTCR)

Bits	Type	Default	Name	Description
[31:2]	RO	0x0	Reserved	Reserved.
[1]	RW	DBG_RECOV_PORST_CFG	DBG_RECOV_PORST_EN	0b0 –**DEVPORESETn** is not asserted when in DBG_RECOV. 0b1 –**DEVPORESETn** is asserted when in DBG_RECOV. This bit should not be modified when the PPU is in DBG_RECOV, or the PPU is in transition, if it is then PPU behavior is UNPREDICTABLE. If DBG_RECOV is not supported this bit is reserved. See *DBG_RECOV and DEVPORESETn* on page 4-18.
[0]	RW	WARM_RST_DEVREQEN_CFG	WARM_RST_DEVREQEN	0b0 – The PPU does not perform a device interface handshake when transitioning between ON and WARM_RST. 0b1 – The PPU performs a device interface handshake when transitioning between ON and WARM_RST. This bit should not be modified when the PPU is in WARM_RST, or if the PPU is performing a transition, otherwise PPU behavior is UNPREDICTABLE. For a Q-Channel PPU this setting disables all Q-Channels for this transition.

5.2.10 Interrupt Mask Register (PPU_IMR)

This register controls the events that assert the interrupt output. Additional event masking controls are in the *Additional Interrupt Mask Register (PPU_AIMR)*, *Input Edge Sensitivity Register (PPU_IESR)*, and the *Operating Mode Active Edge Sensitivity Register (PPU_OPSR)*.

When an interrupt event is masked an occurrence of the event does not set the corresponding bit in the interrupt status register.

For each bit:

- 0b0 – Event enabled.
- 0b1 – Event masked.

Table 5-13 Interrupt Mask Register (PPU_IMR)

Bits	Type	Default	Name	Description
[31:6]	RO	0x0	Reserved	Reserved.
[5]	RW	0x1	LOCKED_IRQ_MASK	Locked event mask. If the locked interrupt event is not supported, then this bit is reserved.
[4]	RW	0x1	EMU_DENY_IRQ_MASK	Emulation transition denial event mask. When both OFF_EMU and MEM_RET_EMU are not supported, or for a Q-Channel PPU, this bit is reserved.
[3]	RW	0x1	EMU_ACCEPT_IRQ_MASK	Emulation transition acceptance event mask. If both OFF_EMU and MEM_RET_EMU are not supported this bit is reserved.
[2]	RW	0x0	STA_DENY_IRQ_MASK	Static transition denial event mask.
[1]	RW	0x1	STA_ACCEPT_IRQ_MASK	Static transition acceptance event mask.
[0]	RW	0x0	STA_POLICY_TRN_IRQ_MASK	Static full policy transition completion event mask.

5.2.11 Additional Interrupt Mask Register (PPU_AIMR)

This register controls the events that assert the interrupt output. Additional event masking controls are in the *Interrupt Mask Register (PPU_IMR*), *Input Edge Sensitivity Register (PPU_IESR)*, and the *Operating Mode Active Edge Sensitivity Register (PPU_OPSR)*.

When an interrupt event is masked an occurrence of the event does not set the corresponding bit in the interrupt status register.

For each bit:

- 0b0 – Event enabled.
- 0b1 – Event masked.

Table 5-14 Additional Interrupt Mask Register (PPU_AIMR)

Bits	Type	Default	Name	Description
[31:5]	RO	0x0	Reserved	Reserved.
[4]	RW	0x1	STA_POLICY_OP_IRQ_MASK	Static operating policy transition completion event status. If the operating policy transition completion event is not supported this bit is reserved.
[3]	RW	0x1	STA_POLICY_PWR_IRQ_MASK	Static power policy transition completion event status. If the power policy transition completion event is not supported this bit is reserved.
[2]	RW	0x1	DYN_DENY_IRQ_MASK	Dynamic transition denial event mask. When no dynamic transitions are supported this field is reserved.
[1]	RW	0x1	DYN_ACCEPT_IRQ_MASK	Dynamic transition acceptance event mask. When no dynamic transitions are supported this field is reserved.
[0]	RW	0x0	UNSPT_POLICY_IRQ_MASK	Unsupported Policy event mask.

5.2.12 Interrupt Status Register (PPU_ISR)

This register contains information about events causing the assertion of the interrupt output. It is also used to clear interrupt events.

A bit set to 0b1 indicates the event asserted the interrupt output. Multiple events can be active at the same time. When an interrupt event is masked an occurrence of that event does not set the status bit.

A write of 0b1 to an event bit clears that event. A write of 0b0 to a bit has no effect. The interrupt output stays HIGH until all status bits in the *Interrupt Status Register (PPU_ISR)* and the *Additional Interrupt Status Register (PPU_AISR)* are 0b0.

When the OTHER_IRQ bit is set, this indicates an event from the *Additional Interrupt Status Register (PPU_AISR)* has caused the interrupt output to be asserted. This bit cannot be cleared by writing to this register. It must be cleared by writing to the active event in the *Additional Interrupt Status Register (PPU_AISR)*.

Note

For P-Channel PPU, the PWR_ACTIVE_EDGE_IRQ field LSB representing **DEVPACTIVE[0]** is reserved. This bit represents OFF and is not typically driven by a component.

Bits for unused channels are the MSBs and are reserved.

Table 5-15 Interrupt Status Register (PPU_ISR)

Bits	Type	Default	Name	Description
[31:24]	RW	0x0	OP_ACTIVE_EDGE_IRQ	Indicates which operating mode **DEVPACTIVE** inputs caused the input edge event. When operating modes are not supported this field is reserved.
[23:19]	RO	0x0	Reserved	Reserved
[18:8]	RW	0x00	PWR_ACTIVE_EDGE_IRQ	Indicates which power mode **DEVACTIVE** inputs caused the input edge event. *For Q-Channel*: There is one bit for each device Q-Channel **DEVQACTIVE**. For example, bit 8 is for the Q-Channel 0 **DEVQACTIVE**, and bit 9 for the Q-Channel 1 **DEVQACTIVE**. For a Q-Channel PPU bits [18:16] are always reserved as only 8 device interface Q-Channels are supported. *For P-Channel*: There is one bit for each bit of the power mode **DEVPACTIVE** group. For example, bit 18 is for **DEVPACTIVE**[10], and bit 17 is for **DEVPACTIVE**[9]. Bit 8 relates to **DEVPACTIVE**[0] and is always 0.
[7]	RO	0x0	OTHER_IRQ	Indicates there is an interrupt event pending in the *Additional Interrupt Status Register (PPU_AISR)*, see page 5-15.
[6]	RW	0x0	Reserved	Reserved.
[5]	RW	0x0	LOCKED_IRQ	Locked event status. If the locked event is not supported this bit is reserved.
[4]	RW	0x0	EMU_DENY_IRQ	Emulated transition denial event status. When both OFF_EMU and MEM_RET_EMU are not supported, or for a Q-Channel PPU, this bit is reserved.
[3]	RW	0x0	EMU_ACCEPT_IRQ	Emulated transition acceptance event status. When both OFF_EMU and MEM_RET_EMU are not supported this bit is reserved.

[2]	RW	0x0	STA_DENY_IRQ	Static transition denial event status.
[1]	RW	0x0	STA_ACCEPT_IRQ	Static transition acceptance event status.
[0]	RW	0x0	STA_POLICY_TRN_IRQ	Static full policy transition completion event status.

5.2.13 Additional Interrupt Status Register (PPU_AISR)

This register contains information about events causing the assertion of the interrupt output. It is also used to clear interrupt events.

A bit set to 0b1 indicates the event asserted the interrupt output. Multiple events can be active at the same time. When an interrupt event is masked an occurrence of that event does not set the status bit.

A write of 0b1 to an event bit clears that event. A write of 0b0 has no effect. The interrupt output stays HIGH until all status bits in the Interrupt Status Register (PPU_ISR) and the Additional Interrupt Status Register (PPU_AISR) are set to 0b0.

When an interrupt status is set to 0b1 in this register it sets the OTHER_IRQ bit in the *Interrupt Status Register (PPU_ISR).* Status bits in this register are only cleared by writing to this register.

Table 5-16 Additional Interrupt Status Register (PPU_AISR)

Bits	Type	Default	Name	Description
[31:5]	RO	0x0	Reserved	Reserved.
[4]	RW	0x0	STA_POLICY_OP_IRQ	Static operating policy transition completion event status. If the operating policy transition completion event is not supported this bit is reserved.
[3]	RW	0x0	STA_POLICY_PWR_IRQ	Static power policy transition completion event status. If the power policy transition completion event is not supported this bit is reserved.
[2]	RW	0x0	DYN_DENY_IRQ	Dynamic transition denial event status. When no dynamic transitions are supported this field is reserved.
[1]	RW	0x0	DYN_ACCEPT_IRQ	Dynamic transition acceptance event status. When no dynamic transitions are supported this field is reserved.
[0]	RW	0x0	UNSPT_POLICY_IRQ	Unsupported Policy event status.

5.2.14 Input Edge Sensitivity Register (PPU_IESR)

This register configures the transitions on the power mode **DEVACTIVE** inputs that generate an Input Edge interrupt event. For each input, there are two registers bits with the following encoding:

- 0b00 – Event masked.
- 0b01 – Event on rising edge.
- 0b10 – Event on falling edge.
- 0b11 – Event on both edges.

When an event is masked an occurrence of the event does not set the corresponding bit in the interrupt status register.

Fields for power mode **DEVACTIVE** inputs that are not supported are reserved.

Note

When configured as a P-Channel, the **DEVACTIVE**[0] edge sensitivity field is reserved. This bit represents OFF and is not typically driven by a component.

Fields for **DEVACTIVE** inputs that are not supported are reserved.

Table 5-17 Input Edge Sensitivity Register (PPU_IESR)

Bits	Type	Default	Name	Description
[31:22]	RO	0x0000	Reserved	Reserved.
[21:20]	RW	0x0	DEVACTIVE10_EDGE	***For Q-Channel***: Reserved. ***For P-Channel***: **DEVACTIVE** 10 edge sensitivity.
[19:18]	RW	0x0	DEVACTIVE09_EDGE	***For Q-Channel***: Reserved. ***For P-Channel***: **DEVACTIVE** 9 edge sensitivity.
[17:16]	RW	0x0	DEVACTIVE08_EDGE	***For Q-Channel***: Reserved. ***For P-Channel***: **DEVACTIVE** 8 edge sensitivity.
[15:14]	RW	0x0	DEVACTIVE07_EDGE	**DEVACTIVE** 7 edge sensitivity.
[13:12]	RW	0x0	DEVACTIVE06_EDGE	**DEVACTIVE** 6 edge sensitivity.
[11:10]	RW	0x0	DEVACTIVE05_EDGE	**DEVACTIVE** 5 edge sensitivity.
[9:8]	RW	0x0	DEVACTIVE04_EDGE	**DEVACTIVE** 4 edge sensitivity.
[7:6]	RW	0x0	DEVACTIVE03_EDGE	**DEVACTIVE** 3 edge sensitivity.
[5:4]	RW	0x0	DEVACTIVE02_EDGE	**DEVACTIVE** 2 edge sensitivity.
[3:2]	RW	0x0	DEVACTIVE01_EDGE	**DEVACTIVE** 1 edge sensitivity.
[1:0]	RW	0x0	DEVACTIVE00_EDGE	***For Q-Channel***: **DEVACTIVE** 0 edge sensitivity. ***For P-Channel***: Reserved.

5.2.15 Operating Mode Active Edge Sensitivity Register (PPU_OPSR)

This register configures the transitions on the operating mode **DEVACTIVE** inputs that generate an Input Edge interrupt event. For each input, there are two registers bits with the following encoding:

- 0b00 – Event masked.
- 0b01 – Event on rising edge.
- 0b10 – Event on falling edge.
- 0b11 – Event on both edges.

When an event is masked an occurrence of the event does not set the corresponding bit in the interrupt status register.

Fields for **DEVPACTIVE** inputs that are not supported are reserved.

Note

If the PPU is configured as a Q-Channel PPU, or when operating modes are not supported, this register is reserved.

Table 5-18 Operating Mode Active Edge Sensitivity Register (PPU_OPSR)

Bits	Type	Default	Name	Description
[31:16]	RO	0x0000	Reserved	Reserved.
[15:14]	RW	0x0	DEVACTIVE23_EDGE	**DEVPACTIVE** 23 edge sensitivity
[13:12]	RW	0x0	DEVACTIVE22_EDGE	**DEVPACTIVE** 22 edge sensitivity
[11:10]	RW	0x0	DEVACTIVE21_EDGE	**DEVPACTIVE** 21 edge sensitivity
[9:8]	RW	0x0	DEVACTIVE20_EDGE	**DEVPACTIVE** 20 edge sensitivity
[7:6]	RW	0x0	DEVACTIVE19_EDGE	**DEVPACTIVE** 19 edge sensitivity
[5:4]	RW	0x0	DEVACTIVE18_EDGE	**DEVPACTIVE** 18 edge sensitivity
[3:2]	RW	0x0	DEVACTIVE17_EDGE	**DEVPACTIVE** 17 edge sensitivity
[1:0]	RW	0x0	DEVACTIVE16_EDGE	**DEVPACTIVE** 16 edge sensitivity

5.2.16 Functional Retention RAM Configuration Register (PPU_FUNRR)

This register controls bits [15:8] of the **PCSMPSTATE** output when in FUNC_RET mode. These outputs are used by the PCSM to configure the RAMs that are retained.

For information on the **PCSMPSTATE** signal enumeration see *PCSMPSTATE Enumeration* on page 4-23.

The enumeration of this register is IMPLEMENTATION DEFINED.

Note

If FUNC_RET is not supported, or FUNC_RET_RAM_REG_CFG is set to 0b0, then this register is reserved.

Table 5-19 Functional Retention RAM Configuration Register (PPU_FUNRR)

Bits	Type	Default	Name	Description
[31:8]	RO	0x0	Reserved	Reserved
[7:0]	RW	0x0	FUNC_RET_RAM_CFG	FUNC_RET RAM configuration bits. If FUNC_RET is not supported, or FUNC_RET_RAM_REG_CFG is set to 0b0, this field is reserved.

5.2.17 Full Retention RAM Configuration Register (PPU_FULRR)

This register controls bits [15:8] of the **PCSMPSTATE** output when in FULL_RET mode. These outputs are used by the PCSM to configure the RAMs that are retained.

For information on the **PCSMPSTATE** signal enumeration see *PCSMPSTATE Enumeration* on page 4-23.

The enumeration of this register is IMPLEMENTATION DEFINED.

Note

If FULL_RET is not supported, or FULL_RET_RAM_REG_CFG is set to 0b0, then this register is reserved.

Table 5-20 Full Retention RAM Configuration Register (PPU_FULRR)

Bits	Type	Default	Name	Description
[31:8]	RO	0x0	Reserved	Reserved
[7:0]	RW	0x0	FULL_RET_RAM_CFG	FULL_RET RAM configuration bits. If FULL_RET is not supported, or FULL_RET_RAM_REG_CFG is set to 0b0, this field is reserved.

5.2.18 Memory Retention RAM Configuration Register (PPU_MEMRR)

This register controls bits [15:8] of the **PCSMPSTATE** output when in MEM_RET mode. These outputs are used by the PCSM to configure the RAMs that are retained.

For information on the **PCSMPSTATE** signal enumeration see *PCSMPSTATE Enumeration* on page 4-23.

The enumeration of this register is IMPLEMENTATION DEFINED.

Note

If MEM_RET is not supported, or MEM_RET_RAM_REG_CFG is set to 0b0, then this register is reserved.

Table 5-21 Memory Retention RAM Configuration Register (PPU_MEMRR)

Bits	Type	Default	Name	Description
[31:8]	RO	0x0	Reserved	Reserved
[7:0]	RW	0x0	MEM_RET_RAM_CFG	MEM_RET RAM configuration bits. If MEM_RET is not supported, or MEM_RET_RAM_REG_CFG is set to 0b0, this field is reserved.

5.2.19 Power Mode Entry Delay Register 0 (PPU_EDTR0)

This register contains the delay values that are used to control the time between the conditions for a power mode transition being met and when the power mode transition occurs. These values are in PPU clock cycles. Details of this function are in *Power Mode Entry* Delay Timers on page 4-14.

Additional delay values can be found in the *Power Mode Entry Delay Register 1 (PPU_EDTR1)* on page 5-20.

Note

Mode entry delay timer functionality is an optional feature. Whether it is supported can be determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

When it is not supported, this register is reserved. If a power mode is not supported the relevant delay value is reserved.

Table 5-22 Power Mode Entry Delay Register 0 (PPU_EDTR0)

Bits	Type	Default	Name	Description
[31:24]	RW	0x00	FULL_RET_DEL	Delay to entering FULL_RET. If FULL_RET is not supported this field is reserved.
[23:16]	RW	0x00	LOGIC_RET_DEL	Delay to entering LOGIC_RET. If LOGIC_RET is not supported this field is reserved.
[15:8]	RW	0x00	MEM_RET_DEL	Delay to entering MEM_RET and MEM_RET_EMU. If MEM_RET is not supported, then this field is reserved.
[7:0]	RW	0x00	OFF_DEL	Delay to entering OFF and OFF_EMU.

5.2.20 Power Mode Entry Delay Register 1 (PPU_EDTR1)

This register contains the delay values that are used to control the time between the conditions for a power mode transition being met and when the power mode transition occurs. These values are in PPU clock cycles. Details of this function are in *Power Mode Entry* Delay Timers on page 4-14.

Additional delay values can be found in the *Power Mode Entry Delay Register 0 (PPU_EDTR0)* on page 5-20.

Note

Mode entry delay timer functionality is an optional feature. Whether it is supported can be determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24.

When it is not supported, this register is reserved. If a power mode is not supported the relevant delay value is reserved.

Table 5-23 Power Mode Entry Delay Register 1 (PPU_EDTR1)

Bits	Type	Default	Name	Description
[31:16]	RO	0x00	Reserved	Reserved
[15:8]	RW	0x00	FUNC_RET_DEL	Delay to entering FUNC_RET. If FUNC_RET is not supported this field is reserved.
[7:0]	RW	0x00	MEM_OFF_DEL	Delay to entering MEM_OFF. If MEM_OFF is not supported this field is reserved.

5.2.21 Device Control Delay Configuration Register 0 (PPU_DCDR0)

This register is used to program device control delay parameters.

These parameters are described in *Device Control Delays* on page 4-16. The default values of these registers are configured by parameters. See *Device Control Delay Parameters* on page 7-3.

Note

Device control delay programmability is an optional feature.

Whether it is supported can be determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24. When device control delay configuration functionality is not supported this register is read only.

The default values for the fields of this register are dependent on PPU configuration options. For more information see *Configuration Options* on page 7-1.

Table 5-24 Device Control Configuration Register 0 (PPU_DCDR0)

Bits	Type	Default	Name	Description
[31:24]	RO	0x000	Reserved	Reserved.
[23:16]	RW	RST_HWSTAT_DLY_CFG	RST_HWSTAT_DLY	Delay from reset de-assertion to HWSTAT update.
[15:8]	RW	ISO_CLKEN_DLY_CFG	ISO_CLKEN_DLY	Delay from isolation enable de-assertion to clock enable assertion.
[7:0]	RW	CLKEN_RST_DLY_CFG	CLKEN_RST_DLY	Delay from clock enable assertion to reset de-assertion.

5.2.22 Device Control Delay Configuration Register 1 (PPU_DCDR1)

This register is used to program device control delay parameters.

These parameters are described in *Device Control Delays* on page 4-16. The default values of these registers are configured by configurable parameters. See *Device Control Delay Parameters* on page 7-3.

Note

Device control delay programmability is an optional feature.

Whether it is supported can be determined by reading the *PPU Identification Register 1 (PPU_IDR1)*, see page 5-24. When device control delay configuration functionality is not supported this register is read only.

The default values for the fields of this register are dependent on PPU configuration options. For more information see *Configuration Options* on page 7-1.

Table 5-25 Device Control Configuration Register 1 (PPU_DCDR1)

Bits	Type	Default	Name	Description
[31:16]	RO	0x0000	Reserved	Reserved.
[15:8]	RW	CLKEN_ISO_DLY_CFG	CLKEN_ISO_DLY	Delay from clock enable de-assertion to isolation enable assertion.
[7:0]	RW	ISO_RST_DLY_CFG	ISO_RST_DLY	Delay from isolation enable assertion to reset assertion.

5.2.23 PPU Identification Register 0 (PPU_IDR0)

This read-only register contains information on the type and number of channels on the device interface and power and operating modes supported.

Additional information on optional features can be found in the *PPU Identification Register 1 (PPU_IDR1)* on page 5-24.

For each *_SPT bit:

- 0b0: Feature not supported.
- 0b1: Feature supported.

The default values for the fields of this register are dependent on PPU configuration options. For further information see *Configuration Options* on page 7-1.

Table 5-26 PPU Identification Register 0 (PPU_IDR0)

Bits	Type	Default	Name	Function
[31:30]	RO	0x0	Reserved	Reserved.
[29]	RO	DYN_WRM_RST_SPT_CFG	DYN_WRM_RST_SPT	Dynamic WARM_RST support. Always set to 0b0 for a Q-Channel PPU.
[28]	RO	DYN_ON_SPT_CFG	DYN_ON_SPT	Dynamic ON support.
[27]	RO	DYN_FUNC_RET_SPT_CFG	DYN_FUNC_RET_SPT	Dynamic FUNC_RET support.
[26]	RO	DYN_FULL_RET_SPT_CFG	DYN_FULL_RET_SPT	Dynamic FULL_RET support.
[25]	RO	DYN_MEM_OFF_SPT_CFG	DYN_MEM_OFF_SPT	Dynamic MEM_OFF support.
[24]	RO	DYN_LGC_RET_SPT_CFG	DYN_LGC_RET_SPT	Dynamic LOGIC_RET support. Always set to 0b0 for a Q-Channel PPU.
[23]	RO	DYN_MEM_RET_EMU_SPT_CFG	DYN_MEM_RET_EMU_SPT	Dynamic MEM_RET_EMU support
[22]	RO	DYN_MEM_RET_SPT_CFG	DYN_MEM_RET_SPT	Dynamic MEM_RET support.
[21]	RO	DYN_OFF_EMU_SPT_CFG	DYN_OFF_EMU_SPT	Dynamic OFF_EMU support.
[20]	RO	DYN_OFF_SPT_CFG	DYN_OFF_SPT	Dynamic OFF support.
[19]	RO	0x0	RESERVED	Reserved.
[18]	RO	STA_DBG_RECOV_SPT_CFG	STA_DBG_RECOV_SPT	DBG_RECOV support. Always set to 0b0 for a Q-Channel PPU.
[17]	RO	0x1	STA_WRM_RST_SPT	WARM_RST support.
[16]	RO	0x1	STA_ON_SPT	ON support.
[15]	RO	STA_FUNC_RET_SPT_CFG	STA_FUNC_RET_SPT	FUNC_RET support.
[14]	RO	STA_FULL_RET_SPT_CFG	STA_FULL_RET_SPT	FULL_RET support.
[13]	RO	STA_MEM_OFF_SPT_CFG	STA_MEM_OFF_SPT	MEM_OFF support.
[12]	RO	STA_LGC_RET_SPT_CFG	STA_LGC_RET_SPT	LOGIC_RET support. Always set to 0b0 for a Q-Channel PPU.
[11]	RO	STA_MEM_RET_EMU_SPT_CFG	STA_MEM_RET_EMU_SPT	MEM_RET_EMU support.
[10]	RO	STA_MEM_RET_SPT_CFG	STA_MEM_RET_SPT	MEM_RET support.
[9]	RO	STA_OFF_EMU_SPT_CFG	STA_OFF_EMU_SPT	OFF_EMU support.
[8]	RO	0x1	STA_OFF_SPT	OFF support.
[7:4]	RO	NUM_OPMODE_CFG	NUM_OPMODE	No. of operating modes supported is NUM_OPMODE + 1.
[3:0]	RO	DEVCHAN_CFG	DEVCHAN	No. of Device Interface Channels • Q-Channel PPU: The number of Q-Channels (minimum of 1). • P-Channel PPU: Set to 0.

5.2.24 PPU Identification Register 1 (PPU_IDR1)

This read-only register contains information on the optional features and configurations that are supported by this PPU.

Additional information on optional features can be found in the *PPU Identification Register 0 (PPU_IDR0)* on page 5-22.

For each *_SPT bit:

- 0b0: Feature not supported.
- 0b1: Feature supported.

The default values for the fields of this register are dependent on PPU configuration options. For more information see *Configuration Options* on page 7-1.

Table 5-27 PPU Identification Register 1 (PPU_IDR1)

Bits	Type	Default	Name	Function
[31:13]	RO	0x0	Reserved	Reserved.
[12]	RO	OFF_MEM_RET_TRANS_CFG	OFF_MEM_RET_TRANS	OFF to MEM_RET direct transition. Indicates if direct transitions from OFF to MEM_RET and from OFF_EMU to MEM_RET_EMU are supported.
[11]	RO	0x0	Reserved	Reserved.
[10]	RO	OP_ACTIVE_CFG	OP_ACTIVE	Operating mode use model for dynamic transitions. See *Operating Mode DEVPACTIVE Inputs* on page 3-24 and Operating Mode Active *Configuration* on page 7-6. When operating modes are not supported this field is reserved.
[9]	RO	STA_POLICY_OP_IRQ_CFG	STA_POLICY_OP_IRQ_SPT	Operating policy transition completion event status. See *Static Policy Transition Completion* on page 4-5.
[8]	RO	STA_POLICY_PWR_IRQ_CFG	STA_POLICY_PWR_IRQ_SPT	Power policy transition completion event status. See *Static Policy Transition Completion* on page 4-5.
[7]	RO	0x0	Reserved	Reserved.
[6]	RO	FUNC_RET_RAM_REG_CFG	FUNC_RET_RAM_REG	Indicates if the PPU_FUNRR register is present or reserved. 0b0 – Reserved. 0b1 – Present.
[5]	RO	FULL_RET_RAM_REG_CFG	FULL_RET_RAM_REG	Indicates if the PPU_FULRR register is present or reserved. 0b0 – Reserved. 0b1 – Present.
[4]	RO	MEM_RET_RAM_REG_CFG	MEM_RET_RAM_REG	Indicates if the PPU_MEMRR register is present or reserved. 0b0 – Reserved. 0b1 – Present.

[3]	RO	0x0	Reserved	Reserved.
[2]	RO	LOCK_CFG	LOCK_SPT	Lock and the lock interrupt event are supported. See *Lock* on page 3-14.
[1]	RO	SW_DEV_DEL_CFG	SW_DEV_DEL_SPT	Software device delay control configuration support.
[0]	RO	PWR_MODE_ENTRY_DEL_CFG	PWR_MODE_ENTRY_DEL_SPT	Power mode entry delay support.

5.2.25 Implementation Identification Register (PPU_IIDR)

This register provides information about the implementer and implementation of the PPU.

Table 5-28 Implementation Identification Register (PPU_IIDR)

Bits	Type	Default	Name	Description
[31:20]	RO	-	PRODUCT_ID	IMPLEMENTATION DEFINED value identifying the PPU part.
[19:16]	RO	-	VARIANT	IMPLEMENTATION DEFINED value used to distinguish product variants, or major revisions of the product.
[15:12]	RO	-	REVISION	IMPLEMENTATION DEFINED value used to distinguish minor revisions of the product.
[11:0]	RO	-.	IMPLEMENTER	Implementer identification. [11:8] The JEP106 continuation code of the implementer. [7] Always 0. [6:0] The JEP106 identity code of the implementer. For an Arm implementation, bits [11:0] are 0x43B.

5.2.26 Architecture Identification Register (PPU_AIDR)

This register identifies the PPU architecture revision.

[7:0] – 0x11: Indicates PPU architecture revision 1.1.

Table 5-29 Architecture Identification Register (PPU_AIDR)

Bits	Type	Default	Name	Description
[31:8]	RO	0x0	Reserved	Reserved.
[7:4]	RO	0x1	ARCH_REV_MAJOR	0x1 – PPU architecture major revision 1. Other Values – Reserved.
[3:0]	RO	0x1	ARCH_REV_MINOR	0x1 – PPU Architecture minor revision 1. Other Values – Reserved.

6 Signal Descriptions

This section describes PPU signals. It contains the following sections:

- *Clock and Reset* on page 6-2.
- *Software Interface* on page 6-3.
- *Device Interface* on page 6-4.
- *Device Control Signals* on page 6-4.
- *PCSM Interface Signals* on page 6-6.

6.1 Clock and Reset Signals

Table 6-1 shows the PPU clock and reset signals.

Table 6-1 Clock and reset signals

Signal	Direction	Description
CLK	Input	Clock
RESETn	Input	Reset

6.2 Software Interface Signals

6.2.1 Bus Interface Signals

The type of bus interface used is IMPLEMENTATION DEFINED.

Table 6-2 shows an example of an APB interface.

Table 6-2 Example APB bus interface signals

Signal	Direction	Description
PADDR	Input	APB Address
PSEL	Input	APB Select
PENABLE	Input	APB Enable
PWRITE	Input	APB Write Select
PREADY	Output	APB Ready
PWDATA[31:0]	Input	APB Write Data
PRDATA[31:0]	Output	APB Read Data

6.2.2 Interrupt Signals

Table 6-3 shows the interrupt signal.

Table 6-3 Interrupt signal

Signal	Direction	Description
IRQ	Output	Interrupt

6.3 Device Interface Signals

The device interface is configured to be either one or more Q-Channels or a single P-Channel. For more information on the Q-Channel and P-Channel interfaces see the *Low Power Interface Specification ARM Q-Channel and P-Channel Interfaces*.

Table 6-4 shows the device interface signals for a Q-Channel PPU, up to 8 device interface Q-Channels can exist per PPU.

Note

It is IMPLEMENTATION DEFINED whether a PPU with multiple Q-Channels uses multi-bit signals or naming modifications to differentiate each channel.

Table 6-4 Q-Channel signals

Signal	Direction	Description
DEVQACTIVE	Input	Q-Channel active.
DEVQREQn	Output	Q-Channel request.
DEVQACCEPTn	Input	Q-Channel accept.
DEVQDENY	Input	Q-Channel deny.

Table 6-5 shows the device interface signals for a P-Channel PPU.

Table 6-5 P-Channel signals

Signal	Direction	Description
DEVPACTIVE[23:0][a]	Input	P-Channel active.
DEVPSTATE[7:0][a]	Output	P Channel state bus.
DEVPREQ	Output	P-Channel request signal.
DEVPACCEPT	Input	P-Channel accept signal.
DEVPDENY	Input	P-Channel deny signal.

[a] This is the maximum width, it can be reduced if not all modes are supported.

6.4 Device Control Signals

Table 6-6 shows the device control signals.

Table 6-6 Device control signals

Signal	Direction	Description
PPUHWSTAT[31:0][a]	Output	PPU hardware status.
DEVCLKEN	Output	Domain clock enable.
DEVEMUCLKEN	Output	Domain emulated mode clock enable.
DEVPORESETn	Output	Domain power on reset.
DEVRETRESETn	Output	Domain retention reset.
DEVWARMRESETn	Output	Domain warm reset.
DEVISOLATEn	Output	Domain isolation control.
DEVEMUISOLATEn	Output	Domain emulated isolation control.

[a] This is the maximum width, it can be reduced if not all modes are supported.

6.5 PCSM Interface Signals

Table 6-7 shows the PCSM interface signals.

Table 6-7 PCSM interface signals

Signal	Direction	Description
PCSMPSTATE[15:0][a]	Output	PCSM P-Channel state bus.
PCSMPREQ	Output	PCSM P-Channel request signal.
PCSMPACCEPT	Input	PCSM P-Channel accept signal.
PCSMMODESTAT[7:0][a]	Input	PCSM power mode status signal.

[a] This is the maximum width, it can be reduced if not all modes are supported.

7 Configuration Options

This section describes the elements of the PPU that can be configured. It contains the following sections:

- *PPU Configuration* on page 7-2.

7.1 PPU Configuration

7.1.1 Device Interface Type

The device interface can be configured appropriately for the attached devices. The configuration options are:

- Q-Channel.
- P-Channel.

If a Q-Channel configuration is selected then the number of Q-Channels, DEVCHAN_CFG, must also be configured. This can be from one to eight channels. Coordination of the Q-Channels is managed within the PPU.

For P-Channel configuration there is a single P-Channel, DEVCHAN_CFG is 0x0.

7.1.2 P-Channel Delays

The device interface and PCSM P-Channels can be configured to set the **PSTATE** value a number of PPU clock cycles before **PREQ** goes HIGH.

For further information see *DEVPSTATE Early Assertion* on page 4-12 and *PCSMPSTATE Early Assertion* on page 4-23.

The values are configured in PPU clock cycles:

- DEV_PREQ_DLY – For the Device Interface P-Channel.
- PCSM_PREQ_DLY – For the PCSM P-Channel.

Valid values are 0 (**PREQ** and **PSTATE** change in the same cycle), 1, 2 and 3.

7.1.3 Default Power Policy

The default power mode, DEF_PWR_POLICY, of the PPU at reset can be configured. The valid options and values are:

- ON: 0b1000.
- OFF: 0b0000.

The default setting for the power mode dynamic transition enable, DEF_PWR_DYN_EN, can also be set.

Valid values are 0 and 1.

7.1.4 Device Control Delay Parameters

The following parameters set the delay, in PPU clock cycles, between the various device control signals as described in *Device Control Delays* on page 4-16.

When the software device control delay configuration support is enabled, these values are the default values of the *Device Control Delay Configuration Register 0*, see page 5-21, and the *Device Control Delay Configuration Register 1 (PPU_DCDR1)*, see page 5-22. When software device control delay configuration is not supported these are the fixed values for the device control delays.

- ISO_CLKEN_DLY_CFG.
- CLKEN_RST_DLY_CFG.
- RST_HWSTAT_DLY_CFG.
- CLKEN_ISO_DLY_CFG.
- ISO_RST_DLY_CFG.

Valid values for each parameter are in the range 0-255.

7.1.5 Retention RAM Configuration Registers

The following parameters sets whether the RAM retention configuration register is present or is reserved. There is a parameter for each power mode which has one of these registers. When the power mode related to the register is not supported then the register is always reserved regardless of these parameters.

- FUNC_RET_RAM_REG_CFG (for PPU_FUNRR).
- FULL_RET_RAM_REG_CFG (for PPU_FULRR).
- MEM_RET_RAM_REG_CFG (for PPU_MEMRR).

For each parameter:

- 0b0 – Register is reserved.
- 0b1 – Register is present.

7.1.6 Transition Control Parameters

The following parameter sets the default value of the WARM_RST_DEVREQEN bit in the *Power Mode Transition Configuration Register (PPU_PTCR)*, see page 5-10.

- WARM_RST_DEVREQEN_CFG.

The following parameter sets the default value of the DBG_RECOV_PORST_EN bit in the *Power Mode Transition Configuration Register (PPU_PTCR)*, see page 5-10.

- DBG_RECOV_PORST_CFG.

7.1.7 Power Mode Support

The ON, OFF and WARM_RST power modes are always supported, then the PPU can optionally support additional power modes.

Table 7-1 shows the additional power modes that can be supported in a P-Channel or a Q-Channel PPU, and the associated parameters.

Table 7-1 P-Channel and Q-Channel power mode support parameters

Power Mode Support	Parameter
FUNC_RET	STA_FUNC_RET_SPT_CFG
MEM_OFF	STA_MEM_OFF_SPT_CFG
FULL_RET	STA_FULL_RET_SPT_CFG
MEM_RET_EMU[a]	STA_MEM_RET_EMU_SPT_CFG
MEM_RET	STA_MEM_RET_SPT_CFG
OFF_EMU[b]	STA_OFF_EMU_SPT_CFG

[a] MEM_RET_EMU can only be supported if MEM_RET and OFF_EMU are also supported, see Table 7-2.

[b] If MEM_RET is supported then OFF_EMU can only be supported if MEM_RET_EMU is also supported, see Table 7-2.

The emulation modes for OFF and MEM_RET can only be supported if all possible emulation modes are supported. There are no dependencies to other power modes. Table 7-2 shows the allowed configurations.

Table 7-2 Allowed configuration of emulation modes

Power Mode Supported			
OFF	**OFF_EMU**	**MEM_RET**	**MEM_RET_EMU**
YES	NO	NO	NO
YES	YES	NO	NO
YES	NO	YES	NO
YES	YES	YES	YES

Table 7-3 shows the additional power modes that can only be supported in a P-Channel PPU, and the associated parameters.

Table 7-3 P-Channel only power mode support parameters

Power Mode Support	Parameter
DBG_RECOV	STA_DBG_RECOV_SPT_CFG
LOGIC_RET	STA_LGC_RET_SPT_CFG

It can also be configured whether dynamic transitions to each power mode, including ON, OFF and WARM_RST, are supported.

Table 7-4 shows the dynamic transitions that can be supported in a P-Channel or a Q-Channel PPU, and the associated parameters.

Table 7-4 P-Channel and Q-Channel power mode dynamic transition configuration

Dynamic Transition Support	Configuration Parameter
Dynamic transitions to ON	DYN_ON_SPT_CFG
Dynamic transitions to FUNC_RET	DYN_FUNC_RET_SPT_CFG
Dynamic transitions to MEM_OFF	DYN_MEM_OFF_SPT_CFG
Dynamic transitions to FULL_RET	DYN_FULL_RET_SPT_CFG
Dynamic transitions to MEM_RET_EMU	DYN_MEM_RET_EMU_SPT_CFG
Dynamic transitions to MEM_RET	DYN_MEM_RET_SPT_CFG
Dynamic transitions to OFF_EMU	DYN_OFF_EMU_SPT_CFG
Dynamic transitions to OFF	DYN_OFF_SPT_CFG

Table 7-5 shows the additional dynamic transitions that can only be supported in a P-Channel PPU, and the associated parameters.

Table 7-5 P-Channel only power mode dynamic transition configuration

Dynamic Transition Support	Configuration Parameter
Dynamic transitions to WARM_RST	DYN_WRM_RST_SPT_CFG
Dynamic transitions to LOGIC_RET	DYN_LGC_RET_SPT_CFG

Valid configuration values for each parameter are 0 (Not Supported) or 1 (Supported).

Note

A power mode must be supported for it to be configured for dynamic transitions.

OFF to MEM_RET Direct Transition Configuration

The OFF_MEM_RET_TRANS_CFG parameter configures if the PPU can perform the following direct transitions:

- OFF to MEM_RET.
- OFF_EMU to MEM_RET_EMU.

It also configures if the **PCSMMODESTAT** input is present.

The valid values for this parameter are:

- 0b0 – The above direct transitions are not allowed, and the **PCSMMODESTAT** input is not present.
- 0b1 – The above direct transitions are allowed, and the **PCSMMODESTAT** input is present.

7.1.8 Operating Mode Support

The following parameters configure the PPU operating mode support.

Note

Operating modes are only supported on a P-Channel PPU (DEVCHAN_CFG = 0). For a Q-Channel PPU all operating mode configuration parameters are ignored.

Default Operating Policy

The default operating mode, DEF_OP_POLICY, at reset can be configured. Valid values are any of the supported operating modes.

The values of the parameter as specified for operating modes in the OP_POLICY column in *Table 3-3 on page 3-7*.

For example, DEF_OP_POLICY = 0b0010, means the default mode is OPMODE_02.

The default setting for the operating mode dynamic transition enable, DEF_OP_DYN_EN, can also be set.

Valid values are 0 and 1.

Operating Mode Active Configuration

The following parameter configures how the operating mode **DEVPACTIVE** inputs request different operating modes.

- OP_ACTIVE_CFG

The valid values for this parameter are:

- 0b0 – Ladder use model.
- 0b1 – Independent use model.

For further information see *Ladder Use Model* on page 3-25 and *Independent Use Model* on page 3-25.

Number of Operating Modes

The following parameter configures the number of operating modes in the PPU.

- NUM_OPMODE_CFG.

The number of operating modes in the PPU is NUM_OPMODE_CFG + 1.

The PPU can support up to 9 operating modes for the ladder use model.

The PPU can support 1, 2, 4, 8, or 16 operating modes for the independent use model

Valid values for NUM_OPMODE_CFG are:

- Ladder use model (OP_ACTIVE_CFG = 0b0): 0 to 8.
- Independent use model (OP_ACTIVE_CFG = 0b1): 0, 1, 3, 7, and 15.

Setting an operating mode value of zero disables operating mode support and there are not any operating mode **DEVPACTIVE** inputs. The following registers / register fields are reserved:

- PPU_PWPR.OP_POLICY.
- PPU_PWPR.OP_DYN_EN.
- PPU_PWSR.OP_STATUS.
- PPU_PWSR.OP_DYN_STATUS.
- PPU_DISR.OP_DEVACTIVE_STATUS.
- PPU_PWCR.OP_DEVACTIVEEN.
- PPU_ISR.OP_ACTIVE_EDGE_IRQ.
- PPU_OPSR (Entire Register).

Table 7-6 shows the valid values and the resultant number of operating modes and operating mode **DEVPACTIVE** inputs.

Table 7-6 Operating mode parameters

NUM_OPMODE_CFG	No. of Operating Modes	No. of operating mode DEVPACTIVE inputs	
		Ladder	Independent
0	1 (Disabled)	0	0
1	2	1	1
2	3	2	Not Valid
3	4	3	2
4	5	4	Not Valid
5	6	5	Not Valid
6	7	6	Not Valid
7	8	7	3
8	9	8	Not Valid
9-14	10-15	Not Valid	Not Valid
15	16	Not Valid	4

Operating Mode PCSM Transition Configuration

The OPMODE_PCSM_SPT_CFG parameter configures if the PPU performs a PCSM handshake for operating mode only transitions.

When PCSM handshakes are disabled for operating mode only transitions the **PSCMPSTATE[7:4]** outputs are not present or are tied to 0b0000. The PCSM holds no context for operating modes in this case.

The valid values for this parameter are:

- 0 – PCSM handshakes are not performed for operating mode only transitions and the **PSCMPSTATE[7:4]** input bits are not present or are tied LOW.
- 1 – PSCM handshakes are performed for operating mode only transitions.

7.1.9 Feature Support

Table 7-7 shows features that can be optionally supported for P-Channel and Q-Channel PPUs and their associated parameters.

Table 7-7 PPU feature parameters

Feature Support	Parameter
Operating policy transition completion event support See *Static Policy Transition Completion* on page 4-5.	STA_POLICY_OP_IRQ_CFG
Power policy transition completion event support. See *Static Policy Transition Completion* on page 4-5.	STA_POLICY_PWR_IRQ_CFG
Lock support. See *Lock* on page 3-14 and *PPU Interrupt* on page 4-4.	LOCK_CFG When this is 0b1 the lock feature and lock interrupt event are present. This parameter is ignored unless dynamic power mode transitions are supported for either MEM_RET or OFF. See *Power Mode Support* on page 7-4.
Software device delay control configuration support.	SW_DEV_DEL_CFG
Power Mode Entry Delay Support.	PWR_MODE_ENTRY_DEL_CFG

8 Appendix: AXI LPI Support

This section describes support on the PPU device interface for the AXI LPI protocol. It has the following section:

- *AXI LPI Support* on page 8-2.

8.1 AXI LPI Support

The AXI LPI interface is supported as a device interface by connecting it to a Q-Channel interface on the PPU. This section describes those connections. Connection of an AXI LPI interface to a P-Channel is not supported.

Note

The PPU does not support using the **CACTIVE** signal as a denial mechanism as described in *AMBA® AXI™ and ACE™ Protocol Specification*.

The AXI LPI connections to the Q-Channel interface are illustrated in Figure 8-1.

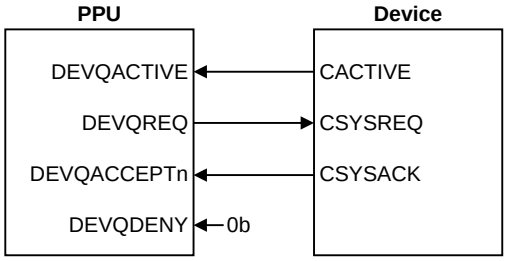

Figure 8-1 AXI LPI to Q-Channel Connections

When a policy other than ON is programmed, the PPU waits until the **DEVQACTIVE** input is LOW and then requests quiescence entry by setting **DEVQREQn** LOW. Figure 8-2 shows a dynamic transition where entry and exit from quiescence is initiated by the **DEVQACTIVE** input.

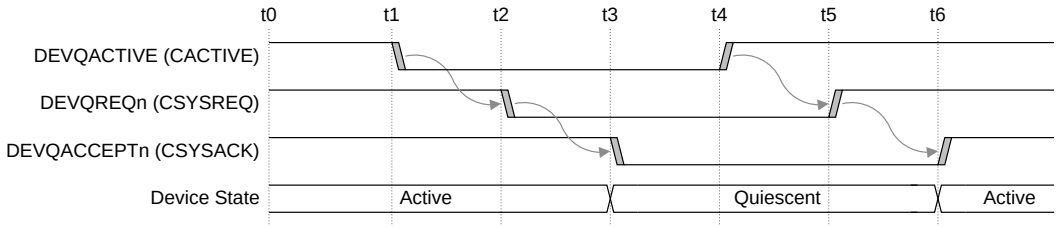

Figure 8-2 PPU device interface : AXI LPI dynamic retention

9 Appendix: Operating Mode Use Model Examples

This section describes some examples for the operating mode use models for the **DEVPACTIVE** inputs. These are illustrative examples and are not intended to be exhaustive. It includes the following sections:

- *Ladder Use Model Example – Cache* Memory on page 9-2.
- *Independent Use Model Example – Multi-threaded Processor* Core on page 9-4.

9.1 Ladder Use Model Example – Cache Memory

In this example, a component has a cache memory partitioned into 4 banks. Each partition can be turned off in turn to reduce the cache size and save power. However, the caches must be turned on and off in a specific order, for example, Cache Bank 0 must be on first, then Cache Bank 1, then 2 and 3.

Table 9-1 Operating mode ladder DEVPACTIVE mode example

Operating Mode	Cache Bank 0	Cache Bank 1	Cache Bank 2	Cache Bank 3	DEVPACTIVE Bit	DEVPSTATE [7:4]
OPMODE_04	ACTIVE	ACTIVE	ACTIVE	ACTIVE	20	0b0100
OPMODE_03	ACTIVE	ACTIVE	ACTIVE	OFF	19	0b0011
OPMODE_02	ACTIVE	ACTIVE	OFF	OFF	18	0b0010
OPMODE_01	ACTIVE	OFF	OFF	OFF	17	0b0001
OPMODE_00	OFF	OFF	OFF	OFF	16	0b0000

The Cache Bank RAM state is described as ACTIVE, rather than ON, as the state of an ACTIVE RAM reflects the current power mode. Table 9-2 shows the actual RAM power mode, as determined by the power mode and the RAM operating mode.

Table 9-2 RAM power state example

	RAM Power Mode in PPU Power Mode				
RAM Mode	**ON**	**FUNC_RET**	**MEM_OFF**	**MEM_RET**	**OFF**
ACTIVE	On	Retention	Off	Retention	Off
OFF	Off	Off	Off	Off	Off

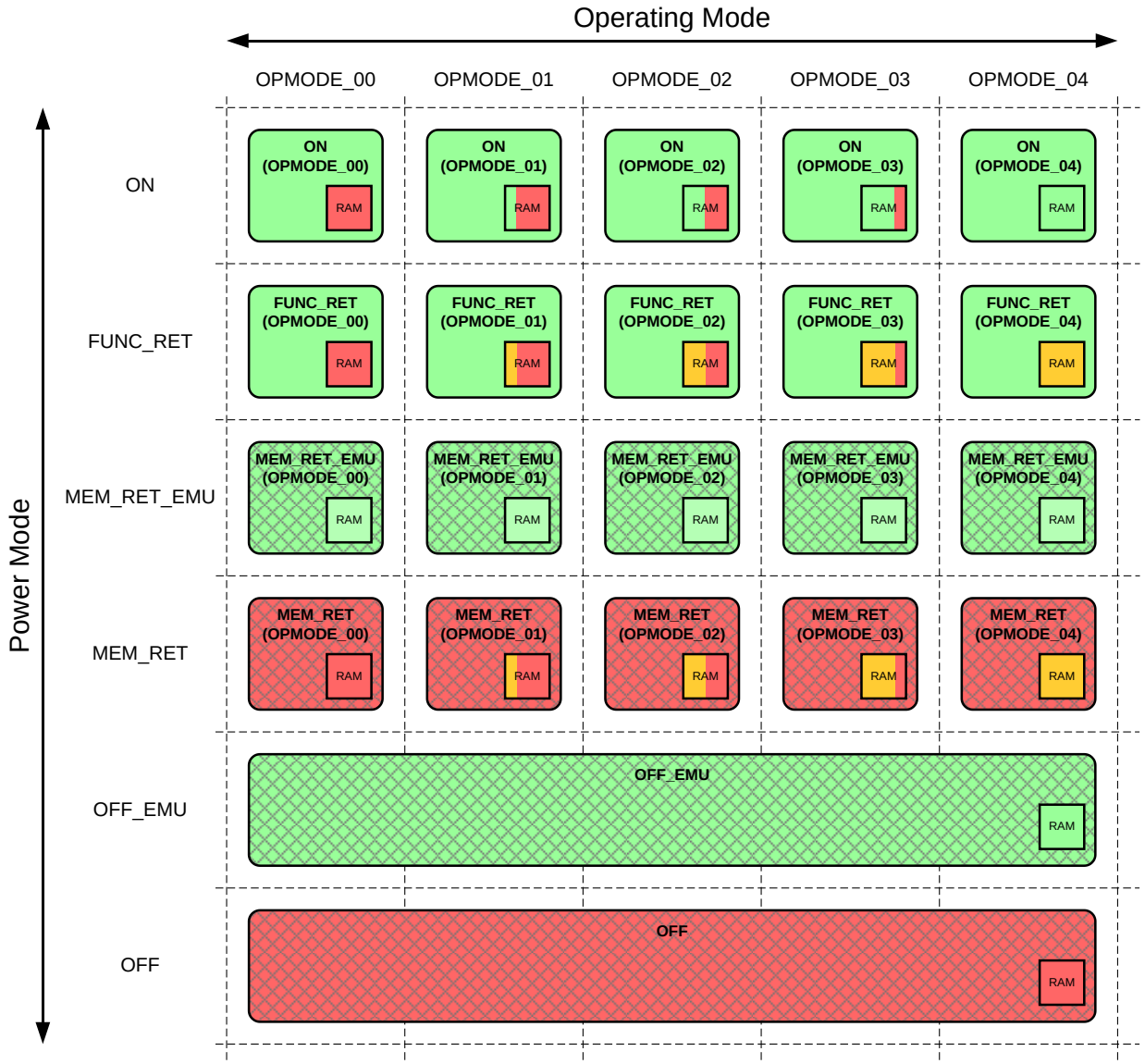

Figure 9-1 – Power and operating mode matrix: Reconfigurable cache example

9.2 Independent Use Model Example – Multi-threaded Processor Core

In this example, we have a processor core with multiple logical threads. Each thread can become idle and request to power off separately, however the physical core power domain is power-managed based on the combined requests from all threads. When wake requests for threads are received at the power controller it must only wake the required thread.

In this case, each thread requests to be active through an operating mode **DEVPACTIVE**. Any combination of threads can be active at the same time. In this case, there is no direct correlation between the operating mode and any single operating mode **DEVPACTIVE** input.

Note

In this example, although changing the operating mode affects the power consumption of the component, by increasing or reducing the number of threads running, it does not require any alteration of anything controlled by the PCSM.

Table 9-3 shows how the operating modes relate to the thread activity.

Table 9-3 Multi-threaded core example operating modes

Operating Mode	Thread 0	Thread 1	DEVPSTATE[7:4]
OPMODE_03	ACTIVE	ACTIVE	0b0011
OPMODE_02	INACTIVE	ACTIVE	0b0010
OPMODE_01	ACTIVE	INACTIVE	0b0001
OPMODE_00	INACTIVE	INACTIVE	0b0000

Each **DEVPACTIVE** requests a resource, the associated thread in this case.

Table 9-4 shows which **DEVPACTIVE** input requests which thread.

Table 9-4 Multi-threaded core example DEVPACTIVE enumeration

DEVPACTIVE	Request
[17]	Thread 1
[16]	Thread 0

Therefore, the combination of **DEVPACTIVE** inputs is used to request an operating mode.

Table 9-5 shows the combinations of operating mode **DEVPACTIVE** inputs and the operating modes they request.

Table 9-5 Multi-threaded core example DEVPACTIVE operating mode requests

DEVPACTIVE[17:16]	Request	Requested Mode
0b11	Thread 0 & Thread 1	OPMODE_03
0b10	Thread 1	OPMODE_02
0b01	Thread 0	OPMODE_01
0b00	No Thread	OPMODE_00

If both operating mode **DEVPACTIVE** inputs are LOW, this is typically in combination with the ON and FULL_RET **DEVPACTIVE** inputs going LOW, allowing the core to go to OFF.

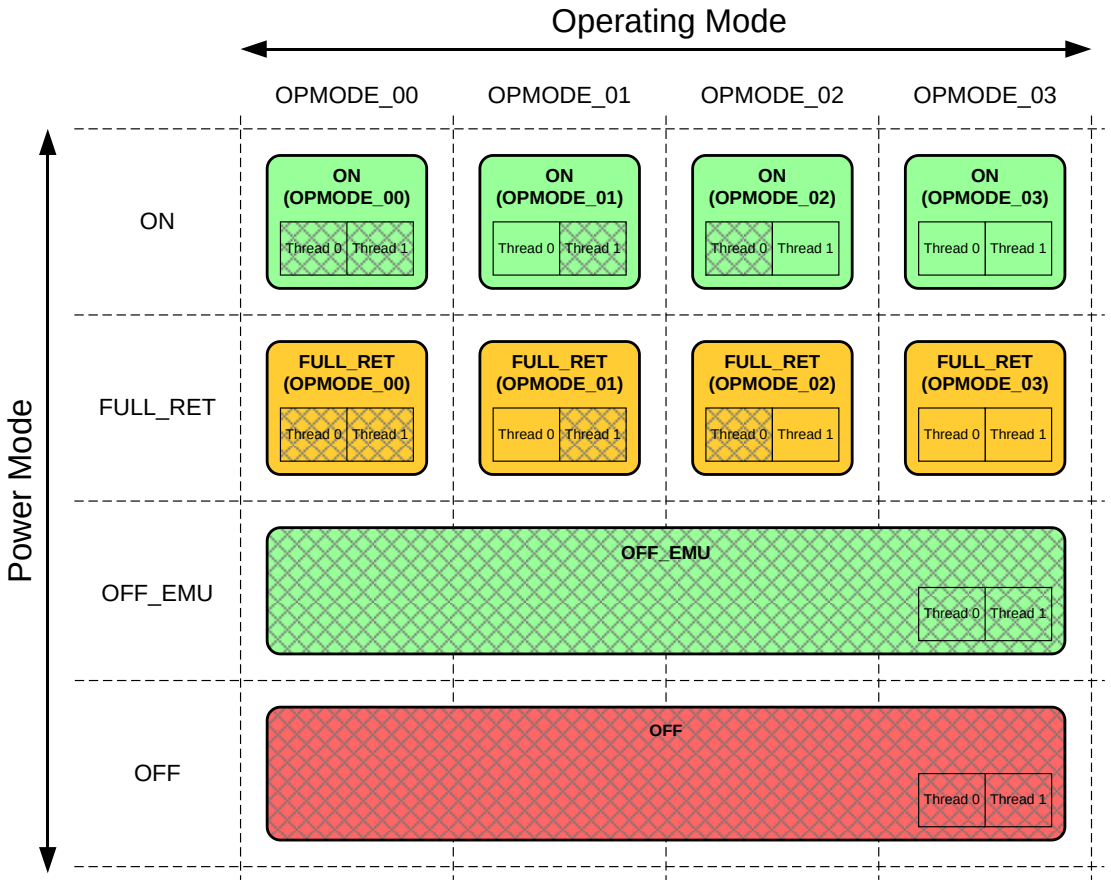

Figure 9-2 – Power and operating mode matrix: Multi-threaded core example

10 Glossary

Current Mode
The current mode of the PPU, reflected in the Power Control Status Register and on the PPUHWSTAT output.

DEVACTIVE
A common term for the device LPI active signals, these are **DEVQACTIVE** for a Q-Channel PPU, and **DEVPACTIVE** for a P-Channel PPU.

DEVREQ
A common term for the device LPI request signals, these are **DEVQREQn** for a Q-Channel PPU and **DEVPREQ** for a P-Channel PPU.

DEVACCEPT
A common term for the device LPI accept signals, these are **DEVQACCEPTn** for a Q-Channel PPU and **DEVPACCEPT** for a P-Channel PPU.

DEVDENY
A common term for the device LPI deny signals, these are **DEVQDENY** for a Q-Channel PPU and **DEVPDENY** for a P-Channel PPU.

Device
A hardware unit that might have a related software driver.

IRQ
Interrupt.

LPI
Low Power Interface; either a Q-Channel or a P-Channel interface.

LSB
Lowest Significant Bit.

PCSM
Power Control State Machine.

P-Channel PPU
A PPU that uses a P-Channel as the device interface.

Power Domain
A collection of design elements within a voltage domain that share common power control. A voltage domain can have one or more power domains.

Power Mode
A combination of the power, reset, clock, and isolation settings.

Power Policy
A software setting that indicates the required power mode, for static transitions, and a minimum power mode, for dynamic transitions.

PPU
Power Policy Unit.

Q-Channel PPU

A PPU that uses between one and eight Q-Channels as the device interface.

RAM

Random Access Memory.

Target Power Mode

For dynamic transitions: The power mode to which the PPU attempts a transition. This is based upon the current power mode, power policy, and **DEVACTIVE** input values.

Voltage Domain

A collection of design elements supplied by a single voltage source. The voltage supply to the domain might be scaled or removed for power or performance reasons.

11 Revisions

This appendix describes the technical changes between released issues of this specification.

Table 11-1 Issue A

Change	Location
First Release	-

Table 11-2 Differences between Issue A and Issue B

Change	Location
Updated figure title to clarify AXI LPI.	*Figure* 8-1 on page 8-2.
Updated PPU_PTCR.DBG_RECOV_PORST_EN field description.	*Table 5-12 Power Mode Transition Configuration Register (PPU_PTCR)* on page 5-11.
Updated PPU_PMER register description to add that register is reserved when both emulated modes are not supported.	*Power Mode Emulation Enable Register (PPU_PMER)* on page 5-4.
Corrected PPU_PWPR.DYN_EN bit name.	*Software Interface* on page 4-2
Updated section title to clarify function.	*Transitions to DBG_RECOV asserting DEVWARMRESETn only* on page 4-36.
Updated section title to clarify function.	*Transitions to DBG_RECOV asserting DEVWARMRESETn only, from OFF and MEM_RET* on page 4-37
Updated figure to add MEM_OFF to OFF_EMU transition.	*Figure* 3-6 on page 3-19.
Updated LOCK_EN description.	*Table 5-2 Power Policy Register (PPU_PWPR)* on page 5-3.
Updated Lock section to clarify lock and unlock scenarios.	*Lock* on page 3-14.
Updated policy transition completion interrupt event to clarify transitions for which it should not be asserted.	*Static Policy Transition Completion* on page 4-5.
Updated dynamic minimum policy interrupt event to clarify transitions for which it should not be asserted.	*PPU Interrupt* on page 4-4.
Updated POLICY to PWR_POLICY and POW_STAT to PWR_STATE to align naming.	*Power Policy Register (PPU_PWPR)* on page 5-3, *Power Status Register (PPU_PWSR)* on page 5-5, and references throughout document.
Removed reference to emulation mode for the Power Status Register.	*Power Status Register (PPU_PWSR)* on page 5-5.
Added LSB (Lowest Significant Bit) to the glossary.	*Glossary* on page 10-1.
Reworded text, no functional change, about transitions which should not include a P-Channel transition.	*P-Channel PPU Power Mode Transitions* on page 3-10.

Corrected the values for the default power mode configuration, DEF_PWR_MODE.	*Default Power Policy* on page 7-2.
Added note that the RAM configuration registers are reserved if the related mode is not supported.	*Functional Retention RAM Configuration Register (PPU_FUNRR)* on page 5-18, *Full Retention RAM Configuration Register (PPU_FULRR)* on page 5-18, and *Memory Retention RAM Configuration Register (PPU_MEMRR)* on page 5-19.
Corrected note on reset levels for DEVREQEN field.	*Power Configuration Register (PPU_PWCR)* on page 5-9.
Corrected Table headers for PPU Identification Registers.	*PPU Identification Register 0 (PPU_IDR0)* on page 5-22, and *PPU Identification Register 1 (PPU_IDR1)* on page 5-24.
Removed the Static Configuration Status Register (PPU_SCSR) as the information is replicated in the PPU Identification Register 0 (PPU_IDR0).	*Register Descriptions* on page 5-3.
Corrected bit numbers in the PPU_AIDR register description.	*Architecture Identification Register (PPU_AIDR)* on page 5-26.
Added DBG_RECOV_PORST_CFG configuration as the default for the PPU_PTCR.DBG_RECOV_PORST.	*Power Mode Transition Configuration Register (PPU_PTCR)* on page 5-10 and Transition Control Parameters on page 7-3.
Updated transitions that do not cause a device interface transition for a P-Channel PPU.	*P-Channel PPU Power Mode Transitions* on page 3-10.
Added parameter to remove RAM configuration registers if not required.	*Functional Retention RAM Configuration Register (PPU_FUNRR)* on page 5-18, *Full Retention RAM Configuration Register (PPU_FULRR)* on page 5-18, *Memory Retention RAM Configuration Register (PPU_MEMRR)* on page 5-19, *PPU Identification Register 0 (PPU_IDR0)* on page 5-22, *Retention RAM Configuration Registers* on page 7-3.
Added starting modes when a PCSM P-Channel handshake is not required for transitions to DBG_RECOV.	*Transitions to DBG_RECOV asserting DEVWARMRESETn only* on page 4-36, and *Transitions to DBG_RECOV asserting DEVPORESETn* on page 4-39.
Corrected sequence in P-Channel transition denial figure.	*Figure* 4-36 on page 4-42.
Modified the Stored Status Register to be reserved for a P-Channel PPU.	*Stored Status Register (PPU_STSR)* on page 5-8.
Added note that emulated event interrupt bits are reserved if both emulated modes are not supported.	*Interrupt Mask Register (PPU_IMR)* on page 5-12 and *Interrupt Status Register (PPU_ISR)* on page 5-14.
Modified wording to better explain emulated mode exceptions.	*Static Power Mode Transitions with a P-Channel PPU* on page 3-13.
Add conditions for moving from EMU to non-EMU power modes.	*Effect of Emulation Enable on Q-Channel PPU Power Mode Transitions* on page 3-12.

Table 11-3 Differences between Issue B and Issue C

Change	Location
Added sections on operating modes	*Operating Mode* on page 2-4, *Operating Modes* on page 3-4, *Operating Mode Values* on page 3-7, *Operating Mode Transitions* on page 3-21, *Operating Mode Transition Rules* on page 3-24, *Operating Mode Transition Sequences* on page 4-43, *Operating Mode Support* on page 7-6, *Programmers Model* on page 5-1

Modified power policy transition completion interrupt to add events for power and operating mode interrupt transitions.	*PPU Interrupt* on page 4-4.
Corrected headers in the Power Mode Transition Sequences section to describe power modes rather than modes.	*Transitions to a Higher Priority Power Mode* on page 4-29, and *Transitions to a Lower Priority Power Mode* on page 4-31.
Changed the name of DEF_PWR_MODE to DEF_PWR_POLICY	*Power Policy Register (PPU_PWPR)* on page 5-3 and *Default Power Policy* on page 7-2
Removed non-allowed transitions from list in power mode entry delay section.	*Power Mode Entry* Delay Timers on page 4-14.
Corrected statement that PPU_PTCR.WARM_RST_DEVREQEN is set to 0b0 by default when it is configurable.	*Transition Specific Device Interface Channel Enables* on page 4-14.
Removed incorrect transition entries of MEM_RET to OFF, and MEM_RET_EMU to OFF_EMU in transition types table.	*Table 4-14 Power mode transition types* on page 4-25
Corrected entry in Table 4-7 from OFF_MEM_RET to MEM_RET_EMU.	*Table 4-7 Clock enable behavior for each power mode* on page 4-17.

Table 11-4 Differences between Issue C and Issue D

Change	Location
Modified the default values for PWR_DYN_STATUS and OP_DYN_STATUS to reflect the default values for PWR_DYN_EN and OP_DYN_EN respectively.	*Table 5-4 Power Status Register (PPU_PWSR)* on page 5-5.
Correct statement for operating mode transition completion interrupt event to state that event occurs when the operating mode matches the operating mode policy.	*Operating Policy Transition Completion* on page 4-6.
Changed OP_STATUS to update when moving to a power mode with no operating mode context.	*Operating Mode Transitions* on page 3-21, *Operating Mode Transition Behavior* on page 3-24, *Transition Denial Policy Reversion* on page 4-3, *Static Transition* Acceptance on page 4-7, *Dynamic Transition Acceptance* on page 4-8, and *Table 5-4 Power Status Register (PPU_PWSR)* on page 5-5.
Clarified the types of transfers the operating mode and power mode transition completion interrupts occur on.	*Power Policy Transition* Completion on page 4-6. *Operating Policy Transition Completion* on page 4-6.
Added clarification on operating mode policy writes whilst in non-operating mode context power modes.	*Operating Mode Transitions* on page 3-21.
Corrected the emulation enable signal name.	*Figure 4-13 Device control transition from ON to OFF_EMU, then to* OFF on page 4-21
Clarified the use of the current mode **DEVPACTIVE** for static and dynamic transitions.	*Static Power Mode Transitions with a P-Channel PPU* on page 3-13. *Dynamic Power Mode Transitions with a P-Channel PPU* on page 3-16.

Updated the rules on locking to state that if locking is enabled during emulation transition the PPU will be locked.	*Locking the PPU* on page 3-15.
Added a note on the special case for Q-Channel transitions between ON and WARM_RST.	*Static Power Mode Transitions* on page 3-12.
Modified the wording to limit to ON how high the power mode will go above the target power mode if it is not supported for dynamic transitions.	*Dynamic Power Mode Transitions with a P-Channel PPU* on page 3-16
Corrected the parameter to configure the number of operating modes to NUM_OPMODE_CFG in all cases.	*Operating Mode Support* on page 7-6. *OP_DEVACTIVEEN default values* on page 5-10. *PPU Identification Register 0 (PPU_IDR0)* on page 5-22.
Removed PPU_PTCR.OP_TRANS from the list of registers removed when operating modes are disabled as it no longer exists as a register field.	*Operating Mode Support* on page 7-6.

Table 11-5 Differences between Issue D and Issue E

Change	Location
Corrected statement on supported PACTIVEs from static to dynamic.	*Figure 3-5 Relevant DEVPACTIVE inputs with dynamic transitions enabled* on page 3-17
Added conditional statements on transitions from OFF to MEM_RET when EMU_EN is set and OFF to MEM_RET direct transitions are enabled.	*Effect of Emulation DEVPACTIVEs on P-Channel PPU Static Power Mode Transitions* on page 3-13
Clarified statement that ON to ON operating mode transitions can occur while a transition to OFF is waiting for the entry delay counter to expire.	*Operating Mode Transition Behavior* on page 3-24