

Arm® Neoverse™ CMN-700 Coherent Mesh Network

Revision: r0p0

Technical Reference Manual

Confidential

arm

Arm® Neoverse™ CMN-700 Coherent Mesh Network

Technical Reference Manual

Copyright © 2020 Arm Limited or its affiliates. All rights reserved.

Release Information

Document History

Issue	Date	Confidentiality	Change
0000-01	20 November 2020	Confidential	First release for r0p0 LAC

Confidential Proprietary Notice

This document is CONFIDENTIAL and any use by you is subject to the terms of the agreement between you and Arm or the terms of the agreement between you and the party authorized by Arm to disclose this document to you.

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

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information: (i) for the purposes of determining whether implementations infringe any third party patents; (ii) for developing technology or products which avoid any of Arm's intellectual property; or (iii) as a reference for modifying existing patents or patent applications or creating any continuation, continuation in part, or extension of existing patents or patent applications; or (iv) for generating data for publication or disclosure to third parties, which compares the performance or functionality of the Arm technology described in this document with any other products created by you or a third party, without obtaining Arm's prior written consent.

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

This document may include technical inaccuracies or typographical errors.

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

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

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

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

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

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20348)

Confidentiality Status

This document is Confidential. This document may only be used and distributed in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Product Status

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

Web Address

developer.arm.com

Progressive terminology commitment

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

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

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

Contents

Arm® Neoverse™ CMN-700 Coherent Mesh Network Technical Reference Manual

Preface

<i>About this book</i>	8
<i>Feedback</i>	11

Chapter 1

Introduction

1.1 <i>About CMN-700</i>	1-13
1.2 <i>Compliance</i>	1-15
1.3 <i>Features</i>	1-16
1.4 <i>Interfaces</i>	1-18
1.5 <i>Configurable options</i>	1-19
1.6 <i>Test features</i>	1-34
1.7 <i>Product documentation and design flow</i>	1-35
1.8 <i>Product revisions</i>	1-37

Chapter 2

Functional description

2.1 <i>Components and structural configuration</i>	2-39
2.2 <i>Clocks and resets</i>	2-73
2.3 <i>Power management</i>	2-83
2.4 <i>Network layer functions</i>	2-95
2.5 <i>Discovery</i>	2-163
2.6 <i>Link layer</i>	2-171
2.7 <i>PCIe integration</i>	2-173

2.8	<i>Reliability, Availability, and Serviceability</i>	2-175
2.9	<i>Transaction handling</i>	2-195
2.10	<i>Processor events</i>	2-207
2.11	<i>Quality of Service</i>	2-208
Chapter 3	Programmers model	
3.1	<i>About the programmers model</i>	3-215
3.2	<i>Register summary</i>	3-218
3.3	<i>Register descriptions</i>	3-249
3.4	<i>CMN-700 programming</i>	3-1238
3.5	<i>CML programming</i>	3-1257
Chapter 4	SLC memory system	
4.1	<i>About the SLC memory system</i>	4-1272
4.2	<i>SLC memory system components and configuration</i>	4-1274
4.3	<i>Error reporting and software-configured error injection</i>	4-1286
4.4	<i>Transaction handling in SLC memory system</i>	4-1287
4.5	<i>HN-F class-based resource allocation and arbitration</i>	4-1295
Chapter 5	Debug Trace and PMU	
5.1	<i>DT system overview</i>	5-1300
5.2	<i>DT usage examples</i>	5-1315
5.3	<i>PMU system overview</i>	5-1320
5.4	<i>CXLA PMU system</i>	5-1321
5.5	<i>Secure debug support</i>	5-1322
Chapter 6	Performance optimization and monitoring	
6.1	<i>Performance optimization guidelines</i>	6-1324
6.2	<i>About the Performance Monitoring Unit</i>	6-1326
6.3	<i>HN-F performance events</i>	6-1327
6.4	<i>RN-I performance events</i>	6-1333
6.5	<i>SBSX performance events</i>	6-1337
6.6	<i>MTSX performance events</i>	6-1341
6.7	<i>HN-I performance events</i>	6-1343
6.8	<i>DN performance events</i>	6-1347
6.9	<i>XP PMU event summary</i>	6-1348
6.10	<i>CXG performance events</i>	6-1349
6.11	<i>Occupancy and lifetime measurement using PMU events</i>	6-1350
6.12	<i>DEVENT</i>	6-1351
Appendix A	Protocol feature compliance	
A.1	<i>AXI/ACE-Lite feature support</i>	Appx-A-1353
A.2	<i>CHI feature support</i>	Appx-A-1355
A.3	<i>CXS property support</i>	Appx-A-1356
A.4	<i>CCIX property support</i>	Appx-A-1357
A.5	<i>CHI feature support for CML</i>	Appx-A-1358
Appendix B	Signal descriptions	
B.1	<i>About the signal descriptions</i>	Appx-B-1361
B.2	<i>Clock and reset signals</i>	Appx-B-1362
B.3	<i>CHI interface signals</i>	Appx-B-1363

<i>B.4</i>	<i>ACE-Lite and AXI Interface signals</i>	Appx-B-1369
<i>B.5</i>	<i>APB interface signals</i>	Appx-B-1385
<i>B.6</i>	<i>Device population signals</i>	Appx-B-1386
<i>B.7</i>	<i>Debug, trace, and PMU interface signals</i>	Appx-B-1387
<i>B.8</i>	<i>DFT and MBIST interface signals</i>	Appx-B-1389
<i>B.9</i>	<i>Clock management signals</i>	Appx-B-1394
<i>B.10</i>	<i>Power management signals</i>	Appx-B-1395
<i>B.11</i>	<i>Interrupt and event signals</i>	Appx-B-1396
<i>B.12</i>	<i>Configuration input signals</i>	Appx-B-1397
<i>B.13</i>	<i>Processor event interface signals</i>	Appx-B-1398
<i>B.14</i>	<i>CGL interface signals</i>	Appx-B-1399
<i>B.15</i>	<i>CXLA configuration interface signals</i>	Appx-B-1405

Appendix C

Revisions

<i>C.1</i>	<i>Revisions</i>	Appx-C-1408
------------	------------------------	-------------

Preface

This preface introduces the *Arm® Neoverse™ CMN-700 Coherent Mesh Network Technical Reference Manual*.

It contains the following:

- [About this book](#) on page 8.
- [Feedback](#) on page 11.

About this book

This book is for the Arm® Neoverse™ CMN-700 Coherent Mesh Network product.

Product revision status

The rxpy identifier indicates the revision status of the product described in this book, for example, r1p2, where:

- rx Identifies the major revision of the product, for example, r1.
- py Identifies the minor revision or modification status of the product, for example, p2.

Intended audience

This book is written for system designers, system integrators, and programmers who are designing or programming a *System-on-Chip* (SoC) that uses Arm® Neoverse™ CMN-700 Coherent Mesh Network.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction

This chapter introduces CMN-700 which is an AMBA 5 CHI interconnect with a customizable mesh topology.

Chapter 2 Functional description

This chapter describes functionality achieved when you design and configure the CMN-700 interconnect and its components.

Chapter 3 Programmers model

This chapter describes the application-level registers and provides an overview for programming the CMN-700 interconnect.

Chapter 4 SLC memory system

This chapter describes the optional SLC memory system which is implemented by HN-Fs in the mesh.

Chapter 5 Debug Trace and PMU

This chapter describes the *Debug Trace* (DT) and *Performance Monitoring Unit* (PMU) features that CMN-700 implements.

Chapter 6 Performance optimization and monitoring

This chapter describes the performance optimization techniques and *Performance Monitoring Unit* (PMU) that System integrators can use to optimize the functionality of the interconnect implementation.

Appendix A Protocol feature compliance

This appendix describes the various features that CMN-700 implements from different protocol and architecture specifications.

Appendix B Signal descriptions

This appendix describes the external I/O signals that CMN-700 implements for connection to other hardware in the system.

Appendix C Revisions

This appendix describes the technical changes between released issues of this manual.

Glossary

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

See the [Arm® Glossary](#) for more information.

Typographic conventions

italic

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

bold

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

`monospace`

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

`monospace`

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

`monospace italic`

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

`monospace bold`

Denotes language keywords when used outside example code.

`<and>`

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

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

SMALL CAPITALS

Used in body text for a few terms that have specific technical meanings, that are defined in the [Arm® Glossary](#). For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Timing diagrams

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

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

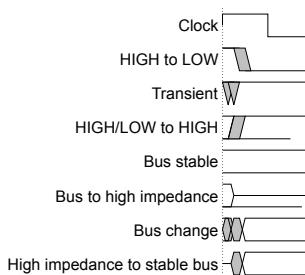


Figure 1 Key to timing diagram conventions

Signals

The signal conventions are:

Signal level

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

Asserted means:

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

Lowercase n

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

Additional reading

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

Arm publications

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

- *AMBA® AXI and ACE Protocol Specification* (IHI 0022)
- *AMBA® CXS Protocol Specification* (IHI 0079)
- *AMBA® Low Power Interface Specification Arm® Q-Channel and P-Channel Interfaces* (IHI 0068)
- *AMBA® 4 AXI4-Stream Protocol Specification* (IHI 0051)
- *AMBA® 5 CHI Architecture Specification* (IHI 0050)
- *Arm® CoreSight™ Architecture Specification* (IHI 0029)
- *Arm® Architecture Reference Manual ARMv7-A and ARMv7-R edition* (DDI 0406)
- *Arm® Architecture Reference Manual Armv8, for Armv8-A architecture profile* (DDI 0487)
- *Arm® Architecture Reference Manual Supplement Memory System Resource Partitioning and Monitoring (MPAM), for Armv8-A* (DDI 0598)
- *Arm® Reliability, Availability, and Serviceability (RAS) Specification Armv8, for the Armv8-A architecture profile* (DDI 0587)
- *Principles of Arm® Memory Maps White Paper* (DEN 0001)

The following confidential books are only available to licensees:

- *Arm® Neoverse™ CMN-700 Coherent Mesh Network Release Note* (PJDOC-1779577084-33602)
- *Arm® Neoverse™ CMN-700 Coherent Mesh Network Configuration and Integration Manual* (102309)
- *Arm® Socrates™ for Neoverse™ CMN-700 User Guide* (102310)
- *Arm® Neoverse™ N1 hyperscale reference design GIC-600 Integration using CMN-600 AXI4-Stream Interfaces White Paper* (PJDOC-1779577084-5931)

Other publications

- *JEDEC Standard Manufacturer's Identification Code*, JEP106, <http://www.jedec.org>.
- *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*, <https://www.ccixconsortium.com/>.

Feedback

Feedback on this product

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

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

Feedback on content

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

- The title *Arm Neoverse CMN-700 Coherent Mesh Network Technical Reference Manual*.
- The number 102308_0000_01_en.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

Arm also welcomes general suggestions for additions and improvements.

————— Note ————

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

Chapter 1

Introduction

This chapter introduces CMN-700 which is an AMBA 5 CHI interconnect with a customizable mesh topology.

It contains the following sections:

- [*1.1 About CMN-700* on page 1-13.](#)
- [*1.2 Compliance* on page 1-15.](#)
- [*1.3 Features* on page 1-16.](#)
- [*1.4 Interfaces* on page 1-18.](#)
- [*1.5 Configurable options* on page 1-19.](#)
- [*1.6 Test features* on page 1-34.](#)
- [*1.7 Product documentation and design flow* on page 1-35.](#)
- [*1.8 Product revisions* on page 1-37.](#)

1.1 About CMN-700

The CMN-700 product is a scalable configurable coherent interconnect that is designed to meet the *Power, Performance, and Area* (PPA) requirements for Coherent Mesh Network systems that are used in high-end networking and enterprise compute applications.

CMN-700 is a scalable mesh interconnect with 1-256 processor compute clusters.

CMN-700 is configurable with the Arm Socrates IP Tooling platform. Socrates is an environment for the configuration of Arm IP. Using Socrates, you can configure the following CMN-700 characteristics:

- Custom interconnect size and device placement.
- Optional *System Level Cache* (SLC). For more information about the features of the SLC memory system, see [4.1 About the SLC memory system on page 4-1272](#).

CMN-700 supports Arm AMBA 5 CHI Issue E, including the following features:

- MakeReadUnique, writes with optional data, and write zero with no data transactions
- Enhanced Exclusive transactions
- Various transaction optimizations and enhancements
- Connection of devices with multiple interfaces
- Extended TxnID and GroupID
- *Distributed Virtual Memory* (DVM) updates
- Memory tagging

CMN-700 provides system-level alignment by providing the following system functionality:

- *Quality of Service* (QoS)
- *Reliability, Availability, and Serviceability* (RAS)
- *Debug and Trace* (DT)

CMN-700 is compatible with the following types of IP:

- *Dynamic Memory Controller* (DMC)
- *Generic Interrupt Controller* (GIC)
- *Memory Management Unit* (MMU)
- Interconnects such as the Arm CoreLink™ NIC-450 Network Interconnect
- Armv8.0, Armv8.2, and Armv8.4 processors

CMN-700 provides an optional *Coherent Multichip Link* (CML) feature. CML is compliant with:

- CCIX standard and allows you to support up to four SoCs in a coherent system

The following table shows the protocol nodes and devices that a system that is built using CMN-700 can contain:

Table 1-1 Supported protocol nodes and devices

Protocol node or device	Description
<i>Fully coherent Requesting Node</i> (RN-F)	<p>A fully coherent master device that supports:</p> <ul style="list-style-type: none"> • CHI Issue A • CHI Issue B • CHI Issue C • CHI Issue D • CHI Issue E <p>————— Restriction —————</p> <p>All RN-Fs must be of the same type.</p>
<i>I/O coherent Requesting Node</i> (RN-I)	An I/O-coherent master device. This CHI bridge device acts as an RN-I proxy for one or more AXI or ACE-Lite master devices that connect to it.

Table 1-1 Supported protocol nodes and devices (continued)

Protocol node or device	Description
<i>I/O coherent Requesting Node with DVM support (RN-D)</i>	An I/O coherent master device that supports acceptance of <i>Distributed Virtual Memory</i> (DVM) messages on the Snoop channel
<i>Fully coherent Home Node (HN-F)</i>	A device that acts as a Home Node for a coherent region of memory. HN-Fs accept coherent requests from RN-Fs and RN-Is, and generates snoops to all applicable RN-Fs in the system as required to support the coherency protocol.
<i>I/O coherent Home Node (HN-I)</i>	A device that acts as a Home Node for the slave I/O subsystem, responsible for ensuring proper ordering of requests targeting the slave I/O subsystem. HN-I supports AMBA AXI and ACE-Lite protocols.
<i>I/O coherent Home Node + DVM Node (HN-D)</i>	A device that includes an HN-I, a <i>Debug Trace Controller</i> (DTC), <i>DVM Node</i> (DN), <i>configuration node</i> (CFG), Global Configuration Slave, and the <i>Power/Clock Control Block</i> (PCCB). ———— Restriction ——— Only one HN-D is allowed per CMN-700 instance. ———— HN-D supports the AMBA AXI, ACE-Lite, ATB, and APB protocols.
<i>I/O coherent Home Node + DTC (HN-T)</i>	An HN-I with a built-in DTC, <i>DVM Node</i> (DN), and ATB. HN-T supports the AMBA AXI and ACE-Lite protocols.
<i>I/O coherent Home Node + DVM + Distributed DVM Node (HN-V)</i>	A device that includes an HN-I and <i>DVM Node</i> (DN). HN-V supports the AMBA AXI and ACE-Lite protocols.
<i>I/O coherent Home Node + PCIe optimization (HN-P)</i>	A device that includes HN-I and dedicated trackers for PCIe peer-to-peer traffic. This device can only be used to connect to PCIe slaves. HN-P supports the AMBA AXI and ACE-Lite protocols.
<i>CHI Slave Node (SN-F)</i>	A device which solely receives CHI commands, limited to fulfilling simple read, write, and CMO requests targeting normal memory
SBSX	A CHI bridge device that converts and forwards simple CHI read, write, and CMO commands to an AXI or ACE-Lite slave memory device
MTSX	A device that connects AXI slave devices that do not support memory tagging to CMN-700
CXG	A <i>CCIX Gateway</i> (CXG) device bridges between CHI and CXL Issue A (CCIX 1.1 port)
CCG	A <i>CXL Gateway</i> (CCG) device bridges between CHI and CXS Issue B (SMP port)

1.2 Compliance

The CMN-700 product is based on Issue E of the AMBA 5 CHI Architecture Specification.

This *Technical Reference Manual* (TRM) complements Architecture Reference Manuals, architecture specifications, protocol specifications, and relevant external standards. It does not duplicate information from these sources.

AMBA® 5 CHI architecture

CMN-700 supports the *AMBA® 5 CHI Architecture Specification* Issue E, and is also backwards compatible with Issue D, Issue C, and Issue B. For more information about compatibility, see [Backward compatible RN-F support on page 2-52](#).

The CMN-700 product implements the following architecture capabilities:

- Fully compliant with CHI interconnect architecture
- Non-blocking coherence protocol
- Packet-based communication
- The following four types of channels:
 - Request (REQ)
 - Response (RSP)
 - Snoop (SNP)
 - Data (DAT)
- Credited end-to-end protocol-layer flow-control with a retry once mechanism for flexible bandwidth and resource allocation
- Integrated end-to-end *Quality-of-Service* (QoS) capabilities

See the *AMBA® 5 CHI Architecture Specification* for more information.

CCIX architecture

The CML CCIX implementation is compliant with *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0* dated September 6, 2019.

1.3 Features

The CMN-700 product provides the following key features:

- Highly scalable mesh network topology configurable up to a 12×12 mesh
- Custom mesh size and device placement
- A programmable *System Address Map* (SAM)
- Up to 256 RN-F interfaces for CHI-based compute clusters, accelerators, graphic processing units, or other cache coherent masters
- Optional *Component Aggregation Layer* (CAL)
- Up to 40 SN interfaces, 80 with CAL
- Up to 40 RN-Is with up to three ACE5-Lite ports each (120 total):

— **Note** —

More devices are supported by adding more levels of interconnect hierarchy to the system. For example, you can use the Arm CoreLink NIC-450 Network Interconnect to add more levels of interconnect hierarchy.

- Option for a second pair of 256-bit REQ, DAT, RSP, and SNP channels, one for each direction. CMN-700 supports either single 256-bit or dual (2x 256-bit) REQ, DAT, RSP, SNP channel configurations for each direction.
- Optional support for non-XY routing algorithm between specified source-target pairs
- Maximum *Physical Address* (PA) width of 52 bits
- DVM message transport between masters
- QoS regulation for shaping traffic profiles
- Configurable QoS override to transactions targeting specific memory regions
- A *Performance Monitoring Unit* (PMU) to count performance-related events
- High-performance distributed SLC and *Snoop Filter* (SF) up to 128 HN-Fs and cache sizes of 0-512MB total:
 - The HN-F includes an integrated *Point-of-Serialization* (PoS) and *Point-of-Coherency* (PoC). The HN-F SLC (also referred to as Agile System Cache) can be used both for compute and I/O caching.
 - SF up to 1024MB of tag RAM for increased coherency scalability consisting of up to 128 partitions (one per HN-F).
- Up to 32, combinations of HN-Is (HN-T, HN-V, HN-P), each with an ACE-Lite master port
- An HN-I that is known as HN-P, which includes PCIe optimizations
- *CHI Memory Tagging Enhancements* (MTE)
- Support for up to 4 devices on edge MXPs
- Option for a single synchronous clock domain or four rectangular asynchronous clock domains
- *Device Credited Slices* (DCSs) used for register slices at device interfaces, allowing flexibility in device placement.
- *Mesh Credited Slices* (MCSs) used for X-Y register slices, allowing flexibility in mesh floorplanning
- *Asynchronous Mesh Credited Slices* (AMCSs) used for asynchronous clock domain crossing
- *Component Aggregation Layer* (CAL) for device interface port expansion
- *CAL Credited Slices* (CCSs), allowing flexibility in mesh floorplanning in configurations that use the CAL
- *On-Chip Memory* (OCM) allowing the creation of CMN-700 systems without physical DDR memory
- RAS features including transport parity, optional data path parity, *Single-Error Correction and Double-Error Detection* (SECDED) ECC, and data poisoning signaling
- Up to 32 CXG or CCG devices
 - CXG devices support CCIX1.1 and have 256-bit or 512-bit CXS Issue A interfaces
 - CCG devices support Coherent Mesh Link SMP (CML_SMP) and have 512-bit CXS Issue B interfaces
- Support for CCIX port-to-port forwarding
- *Address Based Flush* (ABF)
- Way-based SLC partitioning

- Source-based way locking
- *AXI4-Stream* (A4S) support (for GIC traffic only)
- CCIX Slave Agent support for CCIX memory expansion
- Support for AXI loopback signaling
- AXI Utility Bus (AXU) configuration interface for DSU and CHI SN-F interfaces
- Support for AXI read/write burst preservation on PCIe peer-to-peer traffic passing through interconnect targeting local or remote chip

1.4 Interfaces

The following figure shows the interfaces of the CMN-700 product.

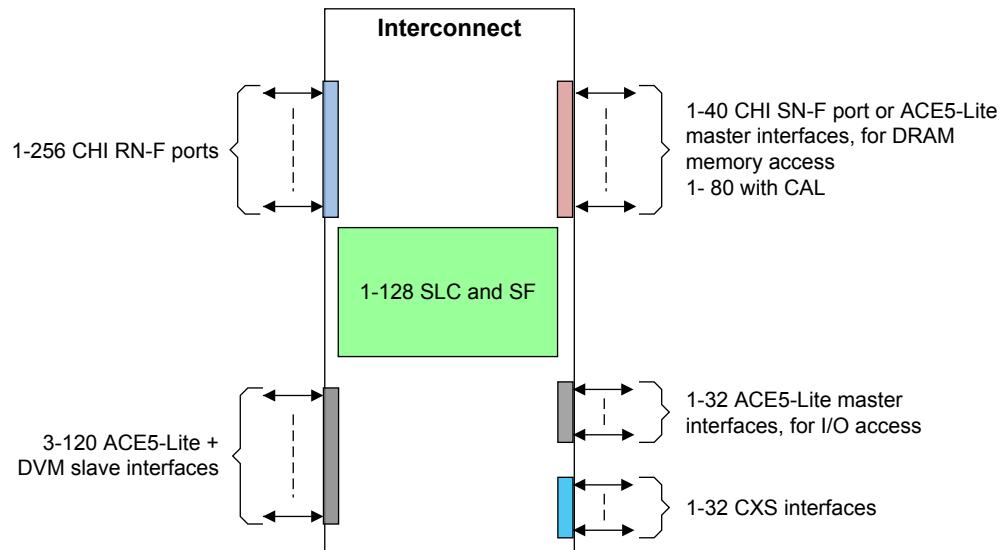


Figure 1-1 CMN-700 interfaces

1.5 Configurable options

The basic structure of CMN-700 is a configurable rectangular grid that is composed of network routers that are known as *Crosspoints* (XPs) and CHI-compliant devices.

Each XP connects horizontally and vertically to other XPs, creating a two-dimensional mesh structure. In a mesh configuration, each XP can have up to four device ports for connecting CHI-compliant devices. In a single-XP configuration, up to six device ports are supported on the XP. For more information, see [Crosspoint on page 2-39](#).

CMN-700 provides several configurable parameters that can be configured to meet system requirements. You can use Socrates to initially auto-populate the devices throughout the mesh. You can then refine the mesh design and device placement using the following guidelines.

CMN-700 is configured in three steps:

1. System component selection. In this step, system components are determined, including:

- Number and type of processors
- I/O interfaces
- Number of HN-Fs
- Amount of SLC
- Memory interfaces

For more information, see [1.5.1 System component selection on page 1-19](#).

2. Mesh sizing and top-level configuration. This step includes specifying the following:

- Number of rows and columns
- Global configuration parameters

For more information, see [1.5.2 Mesh sizing and top-level configuration on page 1-21](#).

3. Device placement and configuration. This step involves:

- Placement of devices and credited repeater slices between XPs based on floorplan needs
- Configuration of devices

For more information, see [1.5.3 Device placement and configuration on page 1-27](#).

1.5.1 System component selection

This section describes CMN-700 system component selection.

Request Nodes

Request Nodes (RNs) reside outside of the mesh and connect to CMN-700 ports.

Requesting masters with coherent caches (processors, GPUs, or processing elements with internal coherent caches) are referred to as RN-F devices. They connect directly to the CMN-700 interconnect mesh using a CHI RN-F port.

I/O-requesting masters without coherent caches connect to CMN-700 RN-I bridge devices using ACE-Lite ports. Examples of I/O-requesting masters include I/O masters, processing elements without internal caches, or processing elements with internal caches that are not hardware coherent. The RN-I bridge device is located between the ACE-Lite interface and the internal CHI interface. Each RN-I bridge device has three ACE-Lite interfaces.

A single I/O-requesting master can be connected directly to a CMN-700 ACE-Lite port. Alternatively, multiple masters can share a single ACE-Lite port by connecting through external AMBA interconnect components. To determine if I/O masters share 1-3 ACE-Lite ports or an RN-I, designers must consider traffic bandwidth requirements and physical floorplan trade-offs.

Home Nodes

In CHI, each byte of address space is assigned to a single *Home Node* (HN). That HN is responsible for handling all memory transactions that are associated with that address.

There are two types of HN devices within the CMN-700 system:

HN-F

HN-F device instances are the HNs for all coherent memory. HN-Fs also support non-coherent memory accesses. Memory that is mapped to an HN-F targets DRAM. Each HN-F can contain an SF and an SLC slice. The amount of SLC required determines the number of HN-Fs.

The total SLC size needed divided by the number of HN-F instances determines the recommended SLC size for each HN-F instance. Generally, each HN-F partition has the same SLC size.

————— **Note** —————

The amount of SLC and number of HN-Fs are configured separately.

————— **Tip** —————

 The optimal total SF size is twice the total exclusive cache size for all RN-Fs. For example, for a 32MB RN-F total cache size, the recommended SF size is 64MB.

HN-I

HN-I device instances are the HNs for all memory that targets an ACE-Lite slave device or subsystem. HN-I does not support coherent memory. However, Cacheable transactions can be sent to HN-I. Each HN-I instance contains a single ACE-Lite master port to send bus transactions to one or more slaves through an AMBA interconnect. The total ACE-Lite master bandwidth requirement, and physical placement of slave peripherals, determines the number of HN-I instances that are needed.

There are HN-I types with extra functionality. These types include:

HN-T HN-I that has a debug trace controller and DVM Node.

CMN-700 can have zero or more HN-I and HN-T instances.

HN-D HN-I that has a debug trace controller, DVM node, and configuration slave.

CMN-700 must have exactly one HN-D instance.

HN-P HN-I that is optimized for peer-to-peer PCIe traffic.

HN-V A device that includes an HN-I and DVM Node (DN).

CML interfaces

The CMN-700 interconnect supports up to 32 CXS (CCIX/CML_SMP port) interfaces. The CXS interface can be either from a CXG device or a CCG device. CXS interface at CXG device is compliant to IssueA of CXS specification, whereas CXS interface at CCG device is compliant to IssueB of CXS specification.

A CXG/CCG device bridges between CHI and CXS, and contains *Request Agent* (RA) proxy and *Home Agent* (HA) proxy functionality. The CXG/CCG device also contains CXS *Link Agent* (LA) functionality, where:

- CXG LA implements CCIX1.1 packet formats (TLPs), is external to the CMN-700 hierarchy
- CCG LA implements CML_SMP flits and is internal CMN-700 hierarchy

Memory interfaces

The CMN-700 interconnect supports two types of memory interface ports:

CHI SN-F port

Connects a native memory controller, such as the CoreLink DMC-620 Dynamic Memory Controller, that complies with:

- CHI Issue C
- CHI Issue D
- CHI Issue E

AXI port

Connects an AXI memory controller using an SBSX bridge. For more information, see [1.5.2 Mesh sizing and top-level configuration on page 1-21](#).

ACE-Lite port

Connects an AXI or ACE-Lite memory controller using either an SBSX or MTSX bridge.

MTSX is only instantiated when the system requires MTE support but the slave AXI or ACE-Lite memory controller does not support MTE. For more information, see [1.5.2 Mesh sizing and top-level configuration on page 1-21](#).

1.5.2 Mesh sizing and top-level configuration

The size of the CMN-700 mesh primarily depends on the number of connected devices.

The minimum number of XPs is half of the number of devices, rounded up. Also, the product of the X and Y mesh dimensions must be greater than or equal to the required number of XPs. For example, if seven XPs are needed, a 2×4 or 4×2 mesh is acceptable.

The following table lists the device types that CMN-700 supports.

Table 1-2 Device types

Device	Name	Description
RN-I	Request Node I/O	A non-caching Request Node that bridges I/O master requests from 1-3 AXI or ACE-Lite interfaces
RN-D	DVM Request Node	An RN-I node that can accept DVM messages on the snoop channel
RNF_CHIB_ESAM	Request Node Full without a built-in SAM, CHI Issue B compliant	CHI Issue B compliant processor, cluster, GPU, or other Request Node with a coherent cache but without a built-in SAM
RNF_CHIC_ESAM	Request Node Full without a built-in SAM, CHI Issue C compliant	CHI Issue C compliant processor, cluster, GPU, or other Request Node with a coherent cache but without a built-in SAM
RNF_CHID_ESAM	Request Node Full without a built-in SAM, CHI Issue D compliant	CHI Issue D compliant processor, cluster, GPU, or other Request Node with a coherent cache but without a built-in SAM
RNF_CHIE_ESAM	Request Node Full without a built-in SAM, CHI Issue E compliant	CHI Issue E compliant processor, cluster, GPU, or other Request Node with a coherent cache but without a built-in SAM
HN-F	Home Node Full	A fully coherent Home Node, typically configured with one or both of SLC and SF
HN-I	Home Node I/O	A device that acts as a Home Node for the slave I/O subsystem, responsible for ensuring proper ordering of requests targeting the slave I/O subsystem. HN-I supports AMBA AXI or ACE-Lite.
HN-T	Home Node I/O with debug trace control	An HN-I with a built-in debug trace controller and DVM Node (DN)
HN-D	DVM Home Node	An HN-I with a built-in debug trace controller, <i>DVM Node (DN)</i> , <i>Configuration Node (CFG)</i> , Global Configuration Slave, and the <i>Power/Clock Control Block (PCCB)</i>

Table 1-2 Device types (continued)

Device	Name	Description
HN-P	Home Node with PCIe optimization	An HN-I with PCIe optimizations, meant for connection to PCIe slaves only
HN-V	Home Node I/O with a DVM Node	An HN-I with a built-in <i>DVM Node</i> (DN)
SN-F	Slave Node	A memory controller consisting of a native CHI-C, CHI-D, or CHI-E SN interface
SBSX	CHI to AXI or ACE-Lite bridge	A CHI to AXI or ACE-Lite bridge that allows an AXI or ACE-Lite memory controller to be connected to CMN-700
MTSX	Memory Tag Slave Interface	A device that connects AXI slave devices that do not support memory tagging to CMN-700
CXG	Internal component of CHI to CXS (CCIX port) bridge	Internal component of CHI to CXS (CCIX port) bridge that enables CML. ————— Note ————— The CXRH device is internal to the CMN-700 hierarchy and includes RA and HA functionality. It connects to an external CXLA device to form a CXG. —————
CCG	CHI to CXS IssueB (CCIX2.0/CXL) Bridge	CHI to CCIX2.0/CXL bridge that enables CML, either for SMP connection or coherence accelerator attachment

The following table shows configurable options for mesh size, component counts, and top-level configuration (including associated parameters).

Table 1-3 Top-level configurable options

Feature	Parameter	Description	Values (Default)	Comments
Mesh dimensions	Mesh X dimension	Number of mesh columns	1-12	Mesh configurations that are not supported: <ul style="list-style-type: none">• 1×2• 2×1 This count is per link and can be different for each link
	Mesh Y dimension	Number of mesh rows	1-12	
	MCSX count	Number of credited slices on an XP-XP mesh link in X dimension	0-4	
	MCSY count	Number of credited slices on an XP-XP mesh link in Y dimension	0-4	
	DCS count	Number of credited slices on a device-XP link	0-4	
	CCS count	Number of credited slices on a CAL-XP link	0-4	

Table 1-3 Top-level configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
Global parameter	RSVDC_METADATA_MO DE_EN	Meta Data Preservation mode enable	0, 1 (False)	Enables DAT RSVDC propagation. For more information, see 2.9.8 DAT RSVDC propagation on page 2-205.
	RSVDC_LOOPBACK_MO DE_EN	RSVDC LoopBack mode enable	0, 1 (False)	-
	RSVDC_LOOPBACK_WID TH	RSVDC LoopBack width	1, 2 (1)	-
	RSVDC_PBHA_MODE_EN	RSVDC PBHA mode enable	0, 1 (False)	-
	RSVDC_PBHA_WIDTH	RSVDC PBHA width	2, 3, 4 (2)	-
	CHI_MPAM_ENABLE	MPAM feature enable	0, 1 (True)	-
	REQ_RSVDC_WIDTH	Width of RSVDC field in REQ flit	0, 4, 8 (4)	-
	REQ_ADDR_WIDTH	Width of ADDR field in REQ flit	44, 48, 52 (48)	REQ_ADDR_WIDTH must be set equal to or greater than PA_WIDTH. Address width of 52 is not supported in a CMN-700 system with RNF_CHIB_ESAM or RNF_CHIC_ESAM devices. For these configurations, CMN-700 supports a maximum REQ_ADDR_WIDTH value of 48.
	PA_WIDTH	System Physical Address width	34, 44, 48, 52 (48)	-
	DATACHECK_EN	Datacheck enable	0, 1 (False)	Data Check refers to data byte parity checking
	NUM_REMOTE_RNF	Number of RNFs for CML configurations on all the remote chips combined.	0-384 (0)	With clustering enabled, local and remote RNFs combined must not exceed 512. When snoop filter clustering is not enabled, local and remote RNF combined must not exceed 128.
	FLIT_PAR_EN	Flit Parity enable	0, 1 (True)	-
	RNSAM_NUM_HTG	Number of hashed regions supported by the RN SAM. This includes SCG, AXID based HNP hashed groups, non-architectural CPAG	0-32 (4)	-
	RNSAM_NP2_EN	Enable non-power of two HNF hashing scheme	0, 1 (False)	-
	RNSAM_HIER_HASH_EN	Enable Hierarchical hashing scheme	0, 1 (False)	-

Table 1-3 Top-level configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
Global parameter	RNSAM_AXID_HASH_EN	Enable AXID based power of two hashing scheme	0, 1 (False)	-
	RNSAMHTG_RCOMP_EN	Enable Range based address comparison for Hashed groups. Program start address and end address for each HTG	0, 1 (False)	-
	RNSAMHTG_RCOMP_LSB	Defines the minimum size of HTG when POR_RNSAMHTG_RCOMP_EN_PARAM = 1, 16 value defines minimum size as 64KB and 26 value defines minimum size as 64MB	16-26 (26)	-
	RNSAM_NUM_ADD_HASHED_TGT	Number of additional hashed target ID's supported by the RN SAM, beyond the local HNF count	0, 2, 4, 8, 16, 32, 64, 96 (0)	For configurations where HN-F CALs are present, see HN-F with CAL support on page 2-120 for further considerations
	RNSAM_NUM_NONHASH_REGION	Number of non-hashed regions supported by the RN SAM	0-64 (8)	-
	RNSAM_NONHASH_RCOMP_MP_EN	Enable Range based address comparison for non-hashed groups. Program start address and end address for each non-hashed groups	0, 1 (False)	-
	RNSAM_NONHASH_RCOMP_LSB	Defines the minimum size of non-hashed group when POR_RNSAM_NONHASH_RCOMP_MP_EN_PARAM = 1, 16 value defines minimum size as 64KB and 26 value defines minimum size as 64MB	16-26 (26)	-
	RNSAM_NUM_CPA_GRP	Number of Aechitectural CPA groups	0-16 (5)	-
	RNSAM_NUM_QOS_REGIONS	Number of memory regions for QoS override	0, 2, 4, 6, 8 (0)	-
	RNSAM_PREFETCH_EN	Enables HNF->MC SAM in the RNSAM which can be used for prefetch type transactions. RNSAM generates a valid SN targetID when this parameter is enabled	0, 1 (True)	-
	RNSAM_PFTGT_NUM_SG	Number of SCG's supported for prefetch transactions by the RNSAM	0-8 (4)	-

Table 1-3 Top-level configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
Global parameter	RNSAM_PFTGT_NUM_NO_NHASH_PSCG	Number of prefetch non-hashed regions supported per System Cache Group by the RNSAM	0-64 (0)	-
	RNSAM_PFTGT_NUM_HTG_PSCG	Number of prefetch HTG regions supported per System Cache Group by the RNSAM	0-8 (0)	-
	RNSAM_PFTGT_DEF_HASHED_GRP_EN	Enable default hashed group for prefetch transactions. To support backward compatible, set this parameter.	0, 1 (True)	-
	RNSAM_FLEX_TGTID_EN	Enables Flexible target ID table base indexes for HNF & CXG target ID's. To support backward compatible, set this parameter to zero	0, 1 (False)	-
	RNSAM_CUSTOM_REGS	Number of customer specific registers for customer implemented logic	0-8 (0)	-
	HNSAM_NUM_NONHASH	Number of non-hashed regions supported by the HNSAM	0-64 (2)	-
	HNSAM_NUM_HTG	Number of HTG regions supported by the HNSAM	0-8 (0)	-
	HNSAM_DEF_HASHED_GRP_EN	Enable default hashed group for HNSAM. To support backward compatible, set this parameter.	0, 1 (True)	-
	HNSAM_RCOMP_EN	Enable Range based address comparison for HNSAM HTG/ Nonhashed groups. Program start address and end address.	0, 1 (False)	-
	HNSAM_RCOMP LSB	Defines the minimum size of HTG when POR_HNSAM_RCOMP_EN_PA RAM = 1, 20 value defines minimum size as 1MB and 26 value defines minimum size as 64MB	20-26 (26)	-
	HNSAM_CUSTOM_REGS	Number of customer specific registers for customer implemented logic	0-2 (0)	-
	EN_2X_DAT_RSP_VC	Enables 2x DAT and RSP VC	0, 1 (False)	-
	EN_2X_REQ_VC	Enables 2x REQ VC	0, 1 (False)	-
	EN_2X_SNP_VC	Enables 2x SNP VC	0, 1 (False)	-

Table 1-3 Top-level configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
Global parameter	MESH2X_DEF_SEL	Selects the default ping-pong scheme for TGTID Lookup in 2xMESH. 0 -> Default Ping-Pong scheme based on Even/Odd XID, 1 -> Default Ping-Pong scheme based on Even/Odd YID	0, 1 (0)	-
	PORTFWD_EN	CCIX Port to Port Forwarding feature enable.	0, 1 (False)	-
	XY_OVERRIDE_CNT	Number of Src-Tgt pairs whose XY route path can be overridden	0, 2, 4, 8, 16 (0)	-
	MXP_MULTIPLE_DTM_EN	Multiple DTM feature enable. This is used if number of device ports on the XP is > 2	0, 1 (False)	-
	RXBUF_NUM_ENTRIES_MCS	Number of entries in the RX Buffer at upload interface of MCSX and MCSY	2-4 (2)	The minimum value of 2 corresponds to a credit return latency of one cycle in the SMXP and one cycle in the MCSX/MCSY
Clock resources	Number of clock inputs	The number of clock inputs in a synchronous or asynchronous mesh	1, 4 (1)	-
Processor resources	Number of RN-Fs	The number of RN-Fs in the system. RN-Fs can be one of the following four types: <ul style="list-style-type: none">• RNF_CHIB_ESAM• RNF_CHIC_ESAM• RNF_CHID_ESAM• RNF_CHIE_ESAM	1-256 without CAL 2-256 with CAL	In a CML system, the maximum number of RN-Fs across all chips is 512. For more information, see 2.4.13 Extended CCIX Requesting Agent ID mechanism for up to 512 RN-Fs on page 2-153 .
I/O resources	Number of RN-Is	The number of RN-I instances in the system	0-40	At least one RN-I or RN-D must be present. The total count of RN-Is and RN-Ds must not exceed 40.
	Number of RN-Ds	The number of RN-D instances in the system	0-40	
	Number of HN-Is	The number of HN-I instances in the system. This count includes HN-V, HN-T, HN-P and the HN-D which is always present.	1-32 1-32 with CAL	CMN-700 supports connection of HN-Is to CAL. When CAL is present, the number of HN-Is must be even.
Debug resources	Number of DTCs	The total number of Debug Trace Controller domains	1-4	The number of DTCs must not exceed the number of HN-Is
System cache	Number of HN-Fs	The total number of HN-F instances in the system. The number of HN-Fs referred to by a given cache group (hashed entry in the SAM) must be a power of two.	1-64 without CAL 2-128 with CAL	For more details, refer to Chapter 4 SLC memory system on page 4-1271 . When CAL is present, the number of HN-Fs must be even.

Table 1-3 Top-level configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
Memory resources	Number of SN-Fs	The number of SN-Fs (CHI interfaces)	0-40	At least one SN-F or SBSX must be present. The total count of SN-Fs and SBSXs must not exceed 40.
	Number of SBSXs	The number of SBSX instances (AXI interfaces)	0-40	CMN-700 supports connection of SBSXs to CAL. When CAL is present, the number of SBSXs must be even.
	Number of CXGs	The number of CXG instances	0-32	-

1.5.3 Device placement and configuration

When the devices are enumerated and the mesh dimensions are determined, the placement of each device or node in the mesh must be specified.

While there are no constraints on the mesh location of a device, floorplanning and performance constraints drive the optimal device placement. This detail is outside the scope of this document.

The following table shows the options that you can use to configure individual CMN-700 devices.

————— Note ————

When CAL is present, all the devices that are connected to it must be configured identically.

Table 1-4 CHI device configurable options

Feature	Parameter	Description	Values (Default)	Comments
RNF-port, SNF-port	POISON	Data poison enable (RN-F port only)	0, 1 (True)	-
	DATACHECK	Data Check enable (RN-F port only)	0, 1 (False)	End-to-end data byte parity enable
	RXBUF_NUM	Number of receive flit buffers inside Booker-CI on this port. To achieve full bandwidth operation, this number must equal the CHI credit return latency (in cycles) for flit transfers from RN-F or SNF to the interconnect.	2-4 (3)	The minimum value of 2 corresponds to a credit return latency of one cycle in the interconnect and one cycle in the RN-F or SNF

Table 1-4 CHI device configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
RN-I, RN-D	AXDATA_WIDTH	Data width on AXI or ACE-Lite interface	128, 256, 512 (128)	-
	NUM_WR_REQ	Number of Write Request Tracker entries	4, 16, 24, 32, 64 (32)	-
	NUM_ATOMIC_BUF	Depth of Atomic data buffers	2, 4, 8, 16, 32 (2)	-
	NUM_RD_REQ	Number of Read Request Tracker entries	4, 32, 64, 96, 128, 256 (32)	If NUM_RD_BUF is 128 or 256, NUM_RD_REQ must be the same value. The number of tracker entries must be the same or larger than data buffer entries. NUM_RD_REQ ≥ NUM_RD_BUF.
	NUM_RD_BUF	Number of Read Data Buffers	4, 8, 16, 24, 32, 64, 96, 128, 256 (24)	This value must be 256 when NUM_RD_REQ is 256 and should be less than or equal to NUM_RD_REQ for all other cases. NUM_RD_BUF > 64 instantiates RAM for data buffer.
	NUM_PREALLOC_RD_BUF	Number of Pre-allocated Read Data Buffers	4, 8, 16, 32 (8)	This value must be ≤ NUM_RD_BUF
	AXDATAPOISON_EN	Data Poison Enable on AXI or ACE-Lite interface	0, 1 (0)	-
	AXLOOPBACK_EN	2-bit Loopback Enable on AXI or ACE-Lite interface	0, 1 (0)	-
	AXMPAM_EN	Enables MPAM feature on AXI or ACE-Lite interfaces	0, 1 (True)	-
	FORCE_RDB_PREALLOC	Force read data buffer pre-allocation	0, 1 (0)	-
	AXID_WIDTH	ID width for slave ports	11, 16, 24, 32 (11)	-

Table 1-4 CHI device configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
HN-F	SLC_SIZE	Size of system cache	0KB, 128KB, 256KB, 512KB, 1MB, 2MB, 3MB, 4MB (2MB)	-
	SF_SIZE	Size of SF tag RAM chosen based on SF_NUM_WAYS as: SF_NUM_WAYS = 16: 512K,1M,2M,4M,8M SF_NUM_WAYS = 32: 1M,2M,4M,8M,16M	512KB, 1MB, 2MB, 4MB, 8MB (4MB)	-
	SF_NUM_WAYS	Number of ways in Snoop Filter cache	16, 32 (16)	-
	SLC_TAG_RAM_LATENCY	Latency of system cache tag RAM	1, 2, 3 (2)	Valid Tag:Data combinations are 1:2, 2:2, 3:3
	SLC_DATA_RAM_LATENCY	Latency of system cache data RAM	2, 3 (2)	Valid Tag:Data combinations are 1:2, 2:2, 3:3
	NUM_ENTRIES_POCQ	Number of entries in the POCQ tracker	16, 32, 64, 128 (32)	-
	SF_RN_ADD_VECTOR_WIDTH	Number of additional bits in the Snoop Filter to track the RN-F's	0-127 (0)	-
	SF_MAX_RNF_PER_CLUSTER	Maximum number of RN-F's per cluster as represented in the SF's RN-F vector	1, 2, 4, 8 (1)	-
	MPAM_NS_PARTID_MAX	Maximum value of non-secure MPAM partitions	1, 2, 4, 8, 16, 32, 64, 128, 256, 512 (64)	-
	MPAM_S_PARTID_MAX	Maximum value of secure MPAM partitions	1, 2, 4, 8, 16, 32, 64, 128, 256, 512 (16)	-
	MPAM_NS_PMG_MAX	Maximum value of non-secure MPAM PMGs	1, 2 (2)	-
	MPAM_S_PMG_MAX	Maximum value of secure MPAM PMGs	1, 2 (2)	-
	MPAM_NUM_CSUMON	Number of CSU monitoring counters	1, 2, 4, 8, 16 (4)	-
HN-I, HN-D, HN-T, HN-V	AXDATA_WIDTH	Data width on AXI or ACE-Lite interface	128, 256, 512 (128)	-
	NUM_AXI_REQS	Number of request tracker entries	8, 32, 64 (32)	-
	AXDATAPOISON_EN	Data poison enable on AXI or ACE-Lite interface	0, 1 (True)	-
	AXMPAM_EN	Enables MPAM feature on AXI or ACE-Lite interfaces	0, 1 (True)	-

Table 1-4 CHI device configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
HN-P	AXDATA_WIDTH	Data width on AXI or ACE-Lite interface	128, 256, 512 (128)	-
	NUM_AXI_REQS	Number of request tracker entries	8, 32, 64 (32)	-
	AXDATAPOISON_EN	Data poison enable on AXI or ACE-Lite interface	0, 1 (True)	-
	AXMPAM_EN	Enables MPAM feature on AXI or ACE-Lite interfaces	0, 1 (True)	-
	WR_NUM_AXI_REQS	Depth of P2P Write Slice OT AXI Request Tracker	32, 64 (32)	-
	RD_NUM_AXI_REQS	Depth of P2P Read Slice OT AXI Request Tracker	64, 128, 256 (64)	-
SBSX	NUM_DART	Number of DART tracker entries	64, 128 (64)	-
	NUM_WR_BUF	Number of write buffers	8, 16 (8)	-
	AXDATA_WIDTH	Data width on AXI or ACE-Lite interface	128, 256 (128)	-
	AXDATAPOISON_EN	Data Poison enable on AXI or ACE-Lite interface	0, 1 (True)	-
	AXMPAM_EN	Enables MPAM feature on AXI or ACE-Lite interfaces	0, 1 (True)	-
	SBSX_CMO_ON_AW	Enables Write Channel CMOs on AXI or ACE-Lite interface	0, 1 (False)	If Enabled, CMOs are sent only on AW channel
	NUM_SBSX_MTU_RDB	Number of Read Data Buffers in SBSX. This is applicable only when SBSX_MTU_EN_PARAM == 1 (POR_CHI_MTE_ENABLE_PARAM == 1 & SBSX_AXMTE_EN_PARAM == 0)	4, 8, 16, 32, 64 (32)	-

Table 1-4 CHI device configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
MTSX	NUM_DART	Number of DART tracker entries	64, 128 (64)	-
	NUM_WR_BUF	Number of write buffers	8, 16 (8)	-
	AXDATA_WIDTH	Data width on AXI or ACE-Lite interface	128, 256 (128)	-
	AXDATAPOISON_EN	Data Poison enable on AXI or ACE-Lite interface	0, 1 (True)	-
	AXMPAM_EN	Enables MPAM feature on AXI or ACE-Lite interfaces	0, 1 (True)	-
	SBSX_CMO_ON_AW	Enables Write Channel CMOs on AXI or ACE-Lite interface	0, 1 (False)	If Enabled, CMOs are sent only on AW channel
	NUM_SBSX_MTU_RDB	Number of Read Data Buffers in SBSX. This is applicable only when SBSX_MTU_EN_PARAM == 1 (POR_CHI_MTE_ENABLE_PARAM == 1 & SBSX_AXMTE_EN_PARAM == 0)	4, 8, 16, 32, 64 (32)	-
	TC_SIZE	Size of Tag Cache	-8, -2, -1, 0, 1, 2 (1)	-
	NUM_TCQ_REQ	Number of TCQ Request Tracker entries	16, 32, 64 (32)	-
	NUM_TCQ_DATA_BUF	Number of TCQ Data Buffer entries	4, 8, 16, 32 (16)	-
	DRAM_ADDR_WIDTH	Width of DRAM Addr Width downstream of MTU block	30, 31, 32, 33, 34 (30)	-

Table 1-4 CHI device configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
CXRH	RA_NUM_REQS	Depth of Request Tracker	64, 128, 256 (256)	-
	RA_NUM_RDBUF	Depth of Read Data Buffer	16, 24, 32 (16)	-
	RA_NUM_WRBUF	Depth of Write Data Buffer	16, 24, 32 (24)	-
	RA_NUM_SNPREQS	Depth of Snoop Tracker	64, 128, 256 (128)	-
	RA_NUM_SNPBUF	Depth of Snoop Data Buffer	16, 24, 32 (32)	-
	HA_NUM_REQS	Depth of request tracker	128, 192, 256 (192)	-
	HA_NUM_WRBUF	Depth of Write Data Buffer	96, 128 (96)	-
	HA_NUM_SNPREQS	Depth of Snoop Tracker	96, 128, 256 (96)	This indicates the number of outstanding snoop request HA can have on CCIX
	HA_NUM_SNPBUF	Depth of Snoop Data Buffer	16, 24, 32 (24)	-
	HA_SSB_DEPTH	This depth indicates the maximum number of remote snoop requests from the local chip that can be sunk at this HA. To avoid deadlock, this queue must sink in all snoops coming to that HA. HN-Fs must ensure that no more than this number is sent to the HA. This value must be at least as large as CXG_HA_NUM_SNPREQS_PARAM	96, 128, 256, 512 (96)	-
	HA_PASS_BUFF_DEPTH	Depth of passive buffer	0, 256, 512 (512)	-
CCG	PCIE_ENABLE	When set, enables PCIe traffic through this CCG	0, 1 (True)	-
	RA_NUM_REQS	Depth of Request Tracker	64, 128, 256 (256)	-
	RA_NUM_RDBUF	Depth of Read Data Buffer	16, 24, 32 (16)	-
	RA_NUM_WRBUF	Depth of Write Data Buffer	16, 24, 32 (24)	-

Table 1-4 CHI device configurable options (continued)

Feature	Parameter	Description	Values (Default)	Comments
CCG continued	RA_NUM_SNPREQS	Depth of Snoop Tracker	64, 128, 256 (128)	-
	RA_NUM_SNPBUF	Depth of Snoop Data Buffer	16, 24, 32 (32)	-
	HA_NUM_REQS	Depth of request tracker	128, 192, 256 (192)	-
	HA_NUM_WRBUF	Depth of Write Data Buffer	64, 96, 128 (64)	-
	HA_NUM_SNPREQS	Depth of Snoop Tracker. This indicates the number of outstanding snoop request HA can have on CCIX.	96, 128, 256 (96)	-
	HA_NUM_SNPBUF	Depth of Snoop Data Buffer	16, 24, 32 (24)	-
	HA_SSB_DEPTH	This depth indicates the maximum number of remote snoop requests from the local chip that can be sunk at this HA. To avoid deadlock, this queue must sink in all snoops coming to that HA. HN-Fs must ensure that no more than this number is sent to the HA. This value must be at least as large as CCG_HA_NUM_SNPREQS_PARAM	96, 128, 256, 512 (96)	-
	HA_REQ_PASS_BUFF_DEPTH	Depth of req channel passive buffer	64, 128, 256, 512 (256)	-
	HA_DAT_PASS_BUFF_DEPTH	Depth of data channel passive buffer	64, 128, 256, 512 (256)	-
	NUM_WR_REQ	Number of PCIe Write Request Tracker entries	4, 16, 24, 32, 64 (32)	-
NUM_RD_REQ	NUM_RD_REQ	Number of PCIe Read Request Tracker entries	4, 32, 64, 96, 128, 256 (32)	If NUM_RD_BUF is 128 or 256, NUM_RD_REQ must be the same value. The number of tracker entries must be the same or larger than data buffer entries. NUM_RD_REQ ≥ NUM_RD_BUF.
	NUM_RD_BUF	Number of Read Data Buffers	4, 8, 16, 24, 32, 64, 96, 128, 256 (32)	This value must be 256 when NUM_RD_REQ is 256 and should be less than or equal to NUM_RD_REQ for all other cases. NUM_RD_BUF > 64 instantiates

1.6 Test features

The CMN-700 product includes several test features.

See the *Arm® Neoverse™ CMN-700 Coherent Mesh Network Configuration and Integration Manual* for information about the test features.

1.7 Product documentation and design flow

The CMN-700 product manuals support the design flow process.

Documentation

The CMN-700 product is supported by the following documentation:

Technical Reference Manual

The *Technical Reference Manual* (TRM) describes the functionality, and how functional options affect the behavior of CMN-700. It is required at all stages of the design flow. The choices that you make in the design flow can mean that some behavior that is described in the TRM is not relevant. If you are programming the CMN-700 product, contact:

- The implementer to determine:
 - The build configuration of the implementation
 - What integration, if any, was performed before implementing the CMN-700 product
- The integrator to determine the pin configuration of the device that you are using.

Configuration and Integration Manual

The *Configuration and Integration Manual* (CIM) describes how to integrate the CMN-700 product into an SoC. It includes a description of the pins that the integrator must tie off to configure the macrocell for the required integration. The CIM also describes:

- The available build configuration options and related issues in selecting them
- How to configure the *Register Transfer Level* (RTL) with the build configuration options
- How to integrate RAM arrays
- How to run test patterns
- The processes to sign off on the configured design.

The Arm product deliverables include reference scripts and information about using them to implement your design. Reference methodology flows supplied by Arm are example reference implementations. Contact your EDA vendor for EDA tool support.

User Guide

The *User Guide* describes how to use Socrates to configure and integrate a custom mesh interconnect.

————— Note —————

The User Guide is part of the productSocrates download bundle.

Safety Manual

The Safety Manual provides additional information on specific features of CMN-700 that are relevant to Functional Safety. This information is important for SoC integrators whose final designs target applications where Functional Safety is a concern.

Development Interface Report

The *Development Interface Report* (DIR) describes the activities conducted by Arm that are related to the safety architecture of CMN-700.

Design flow

CMN-700 is delivered as synthesizable RTL. Before it can be used in a product, it must go through the following processes:

Implementation

The implementer configures and synthesizes the RTL to produce a hard macrocell. This process includes integrating RAMs into the design.

Integration

The integrator connects the implemented design into an SoC. This process includes connecting a memory system and peripherals.

Programming

Programming is the last process. The system programmer develops the software that is required to configure and initialize the CMN-700 product, and tests the required application software.

Each process can:

- Be performed by a different party
- Include implementation and integration choices that affect the behavior and features of the CMN-700 product

The operation of the final device depends on:

Build configuration

The implementer chooses the options that affect how the RTL source files are pre-processed. These options usually include or exclude logic that affects one or more of the area, maximum frequency, and features of the resulting macrocell.

Configuration inputs

The integrator configures some features of the CMN-700 product by tying inputs to specific values. These configurations affect the start-up behavior before any software configuration is made. They can also limit the options available to the software.

Software configuration

The programmer configures the CMN-700 product by programming particular values into registers. The register configuration affects the behavior of the CMN-700 product.

Note

This manual refers to **IMPLEMENTATION DEFINED** features that are applicable to build configuration options. A reference to a feature that is included means that the appropriate build and pin configuration options are selected. A reference to an enabled feature means one that has also been configured by software.

1.8 Product revisions

This section describes the differences in functionality between successive product revisions of the CMN-700 product.

r0p0

First release.

Chapter 2

Functional description

This chapter describes functionality achieved when you design and configure the CMN-700 interconnect and its components.

It contains the following sections:

- [*2.1 Components and structural configuration* on page 2-39.](#)
- [*2.2 Clocks and resets* on page 2-73.](#)
- [*2.3 Power management* on page 2-83.](#)
- [*2.4 Network layer functions* on page 2-95.](#)
- [*2.5 Discovery* on page 2-163.](#)
- [*2.6 Link layer* on page 2-171.](#)
- [*2.7 PCIe integration* on page 2-173.](#)
- [*2.8 Reliability, Availability, and Serviceability* on page 2-175.](#)
- [*2.9 Transaction handling* on page 2-195.](#)
- [*2.10 Processor events* on page 2-207.](#)
- [*2.11 Quality of Service* on page 2-208.](#)

2.1 Components and structural configuration

When configuring CMN-700, the components you select depend on the required functionality. The selected components, topology, and other configurable options affect the overall structure of CMN-700 and the associated behavior.

This section contains the following subsections:

- [2.1.1 Components](#) on page 2-39 describes the individual interconnect components that form CMN-700.
- [2.1.2 System configurations](#) on page 2-53 describes the possible system-level interconnect topologies, including example interconnect configurations.
- [2.1.3 CML system configurations](#) on page 2-58 describes the possible CML system topologies, including example configurations and information about specific CML options.
- [2.1.4 Structural configuration and considerations](#) on page 2-63 describes optional configurations that impact the structure and behavior of CMN-700.

2.1.1 Components

CMN-700 is made up of various types of devices, including router modules, CHI nodes, and bridges. The components that you need depend on the requirements of your system and some are optional or only used if certain requirements are met.

CMN-700 can be integrated into a complete SoC system that includes devices that are not described in this section.

Crosspoint

The *crosspoint* (XP) is a switch or router logic module. It is the fundamental component building block of the CMN-700 transport mechanism.

The CMN-700 mesh interconnect is built using a set of XP modules. The XP modules are arranged in a two-dimensional rectangular mesh topology. Each XP can connect to up to four neighboring XPs using mesh ports, that are shown as dashed lines in the following figure. Each XP also has two device ports for connecting devices, P0 and P1. Depending on the configuration, the XP can have a maximum of:

- 4 device ports for Mesh configurations
- 6 device ports for single MXP configurations

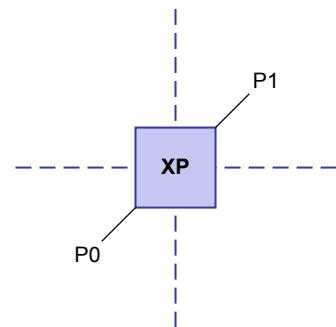


Figure 2-1 Crosspoint (XP)

CMN-700 also supports extra device ports on an MXP. When using extra device ports in a mesh configuration, you can have up to four device ports per XP, as the following figure shows.

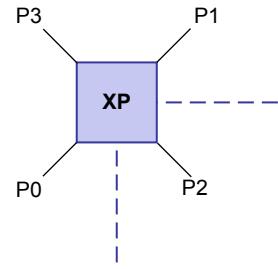


Figure 2-2 Crosspoint structure in mesh configuration with extra device ports

When using a single-XP configuration with extra device ports, you can have up to six device ports on the XP, as the following figure shows.

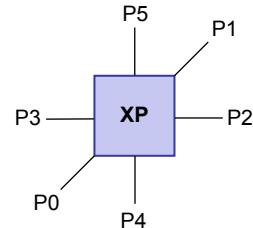


Figure 2-3 Crosspoint structure in single-XP configuration with extra device ports

Each XP supports four CHI channels for transporting flits across the mesh from a source device to a destination or target device:

- *Request* (REQ)
- *Response* (RSP)
- *Snoop* (SNP)
- *Data* (DAT)

The maximum size for the CMN-700 mesh is 144 XPs arranged in a 12×12 grid. Each XP in the grid is referenced using an (X,Y) coordinate system. (0,0) represents the bottom-left corner, and a maximum coordinate of (11,11) represents the upper-right corner. The following figure shows the maximum 12×12 mesh configuration with some (X,Y) coordinate values.

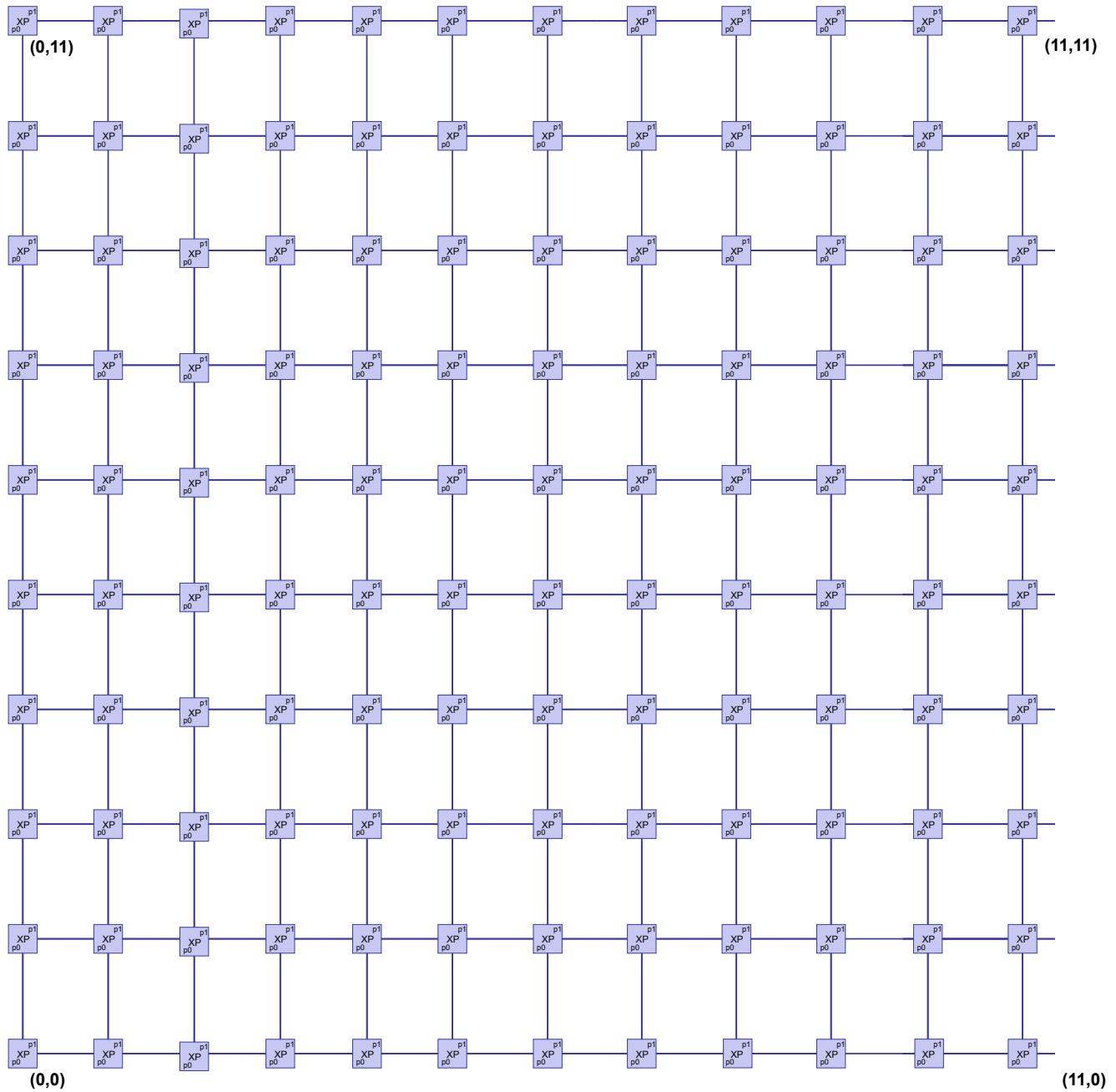


Figure 2-4 12 × 12 maximum mesh configuration

The following figure shows an example 6×6 mesh configuration, with devices attached to XP ports.

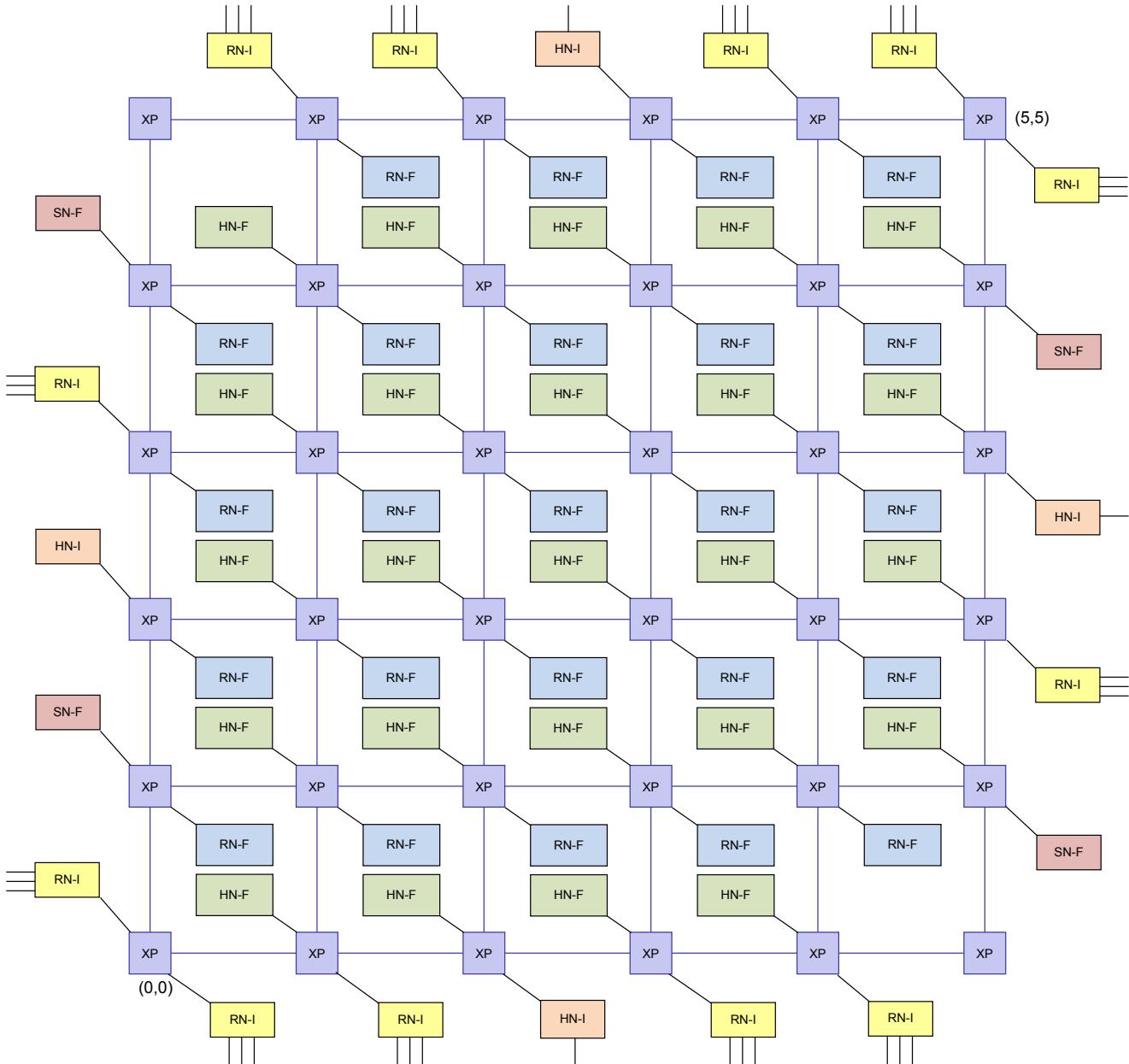


Figure 2-5 Example 6×6 mesh configuration

————— Note —————

The x and y coordinates of an XP are also known as the XID and YID respectively.

I/O coherent Request Node

The *I/O-coherent Request Node* (RN-I) connects I/O-coherent AMBA masters to the rest of the CMN-700 system.

An RN-I bridge includes three ACE-Lite or ACE-Lite-with-DVM slave ports.

The RN-I bridge can act as a proxy only for masters that do not contain hardware-coherent caches. There is no capability to issue snoop transactions to RN-Is.

Fully coherent Home Node

The *Fully coherent Home Node* (HN-F) is responsible for managing part of the address space.

The HN-F consists of the following:

System Level Cache

The *System Level Cache* (SLC) is a last-level cache. The SLC allocation policy is exclusive for data lines, except where sharing patterns are detected and pseudo-inclusive for code lines, as indicated by the RN-Fs. All code lines can be allocated into the SLC on the initial request.

When MTE is enabled, SLC stores data and tags.

Combined PoS/PoC

The combined *Point-of-Serialization/Point-of-Coherency* (PoS/PoC) is responsible for the ordering of all memory requests sent to the HN-F. Ordering includes serialization of multiple outstanding requests and actions to the same line, and request ordering as required by the RN-F.

Snoop Filter

The *Snoop Filter* (SF) tracks cachelines that are present in the RN-Fs. It reduces snoop traffic in the system by favoring directed snoops over snoop broadcasts when possible. This approach substantially reduces the snoop response traffic that might otherwise be required.

Each HN-F in the system is configured to manage a specific portion of the overall address space.

The entire DRAM space is managed through the combination of all HN-Fs in the system.

Note

The HN-F is architecturally defined to manage only well-behaved memory. Well-behaved memory refers to memory without any possible side effects. The HN-F includes microarchitectural optimizations to exploit this architectural guarantee.

I/O coherent Home Node

The *I/O coherent Home Node* (HN-I) is a Home Node for all CHI transactions targeting AMBA slave devices.

The HN-I acts as a proxy for all the RNs of CMN-700, converting CHI transactions to ACE5-Lite transactions. The HN-I includes support for the correct ordering of Arm device types.

The HN-I does not support caching of any data read from or written to the downstream ACE5-Lite I/O slave subsystem. Any cacheable request that is sent to the HN-I does not result in any snoops being sent to RN-Fs in the system. Instead, the request is converted to the appropriate ACE5-Lite read or write command and sent to the downstream ACE5-Lite subsystem.

Caution

If an RN-F caches data that is read from or written to the downstream ACE5-Lite I/O slave subsystem, coherency is not maintained. Any subsequent access to that data reads from or writes to the ACE5-Lite I/O slave subsystem directly, ignoring the cached data.

I/O coherent Home Node with PCIe optimization

The *I/O coherent Home Node with PCIe optimization* (HN-P) is a device that includes the HN-I functionality and dedicated trackers for PCIe peer-to-peer traffic.

HN-P can only be used to connect to PCIe slaves.

SBSX

The *AMBA 5 CHI to ACE5-Lite bridge* (SBSX) enables an ACE5-Lite slave device such as a CoreLink DMC-400 Dynamic Memory Controller, to be used in a CMN-700 system.

MTSX

The *Memory Tag Slave Interface* (MTSX) node connects AXI slave devices without memory tagging support to a CHI-E interconnect that requires MTE support.

MTSX contains two internal elements, an SBSX element and a *Memory Tag Unit* (MTU) element. The MTU element is made up of two parts:

- *Tag Control Queue* (TCQ)
- *Optional Tag Cache* (TC)

In the HN-F SAM, MTSX can only be targeted using non-hashed SN target-based SAM programming.

For more information about the MTSX, see the following sections:

- [2.9.6 MTSX functionality](#) on page 2-204 for information about the functionality of the MTSX.
- [2.8.8 MTU error handling](#) on page 2-188 for information about error handling in the MTSX.
- [3.4.7 MTSX programming](#) on page 3-1245 for information about programming the MTSX.
- [6.6 MTSX performance events](#) on page 6-1341 for information about the MTSX performance monitoring events.

CXG

A CXG device bridges between CHI and CXS IssueA (CCIX1.1 port).

A CXG device contains:

- *CCIX Request Agent* (CXRA) proxy and *CCIX Home Agent* (CXHA) proxy functionality
- *CXS Link Agent* (CXLA) functionality which is external to the CMN-700 hierarchy

CCG

A CCG bridges between CHI and CXS IssueB, which carries either CML_SMP.

The CCG includes:

- *Request Agent* (CCRA) proxy and *Home Agent* (CCHA) proxy functionality, which translate CHI to CML_SMP
- *CXS Link Agent* (CCLA) which implements Upper Link Layer functionality of CCIX2.0 or CXL Data Link Layer, and sits within the CCG block, which is internal to the CMN-700 hierarchy

Note

Optionally, an I/O coherent Requesting Node (RN-I) can be included in CCG block to handle PCIe traffic targeting remote memory or device through either fully coherent or I/O coherent Home Nodes

Configuration node

The *configuration node* (CFG) is co-located with the HN-D node and handles various CMN-700 configuration, control, and monitoring features.

The CFG carries out the following functions:

- Configuration accesses
- Error reporting and signaling
- Interrupt generation
- Centralized debug and PMU support

The CFG includes the following elements:

- Ports to collect error signals from CHI components within CMN-700
- A configuration bus which connects to all the nodes to handle internal configuration register reads and writes
- A dedicated APB interface for configuration accesses

The CFG does not have a dedicated CHI port. It shares a device port with the HN-D node in the mesh.

Power/Clock Control Block

The *Power/Clock Control Block* (PCCB), co-located with the HN-D node, provides separate communication channels. These channels pass information about the power and clock management between the SoC and the network.

The PCCB acts as an aggregator to convey information between the SoC and the other CMN-700 components, in the following manner:

1. The PCCB receives transaction activity indicators from other relevant CMN-700 components and conveys that information to the external power and clock control units.
2. The PCCB receives power or clock control management requests from the external power or clock control units. Where applicable, it conveys that request to the relevant CMN-700 components.
3. The PCCB waits for the appropriate responses from the relevant CMN-700 components, and conveys an aggregated response to the external power and clock control units.

The PCCB does not have a dedicated CHI port. It shares a device port with the HN-D node in the mesh.

If you configure CMN-700 to have multiple asynchronous clock domains, then the PCCB drives one clock signal to each clock domain. For more information about asynchronous clock domain support, see [2.2.1 Clock domain configurations on page 2-73](#).

System Address Map

All CHI commands must include a fully resolved network address. The address must include a source and target ID. Target IDs are acquired by passing a request address through a *System Address Map* (SAM), which effectively maps a memory or I/O address to the target device.

The SAM functionality is required for each requesting device. The SAM consists of two logical units:

RN	Allows each RN to map addresses to HN-F, HN-I, HN-D, and HN-T target IDs. The RN SAM supports generation of <i>Memory Controller</i> (MC) target IDs, which can be used to issue PrefetchTgt operations from the RN directly to the MC.
RN	Allows each RN to map addresses to HN-F, HN-I, HN-D, HN-T, and HN-P target IDs. The RN SAM supports generation of <i>Memory Controller</i> (MC) target IDs, which can be used to issue PrefetchTgt operations from the RN directly to the MC.
HN-F	Maps addresses to MC target IDs.

CMN-700 has software-configurable SAM blocks which allow a single implementation of CMN-700 to support programmable mappings of addresses to HNs and SNs.

The SAM functionality is required for each requesting device.

Debug and Trace Controller

The *Debug and Trace Controller* (DTC) controls distributed *Debug and Trace Monitors* (DTM) and generates time stamped trace using the ATB interface.

The DTC performs the following functions:

- Generates event or PMU-based interrupts
- Receives packets from DTM and packs them into ATB format trace
- Time stamps trace with SoC timer input
- Generates alignment sync for the ATB trace output
- Handles ATB flush requests
- Handles debug and Secure debug external requests
- Provides a consistent view of distributed and central PMU counters
- Handles PMU snapshot requests
- Generates interrupt **INTREQPMU** assertion on overflow of PMU counters

Important

Debug and Trace functionality must be disabled during mission critical operation.

QoS regulator

CMN-700 supports end-to-end *Quality-of-Service* (QoS) which guarantees using QoS mechanisms that are distributed throughout the system.

The QoS provision uses the QoS field in each RN request packet to influence arbitration priority at every QoS decision point. The QoS field is then propagated through all secondary packets issued by a request packet. RNs must either:

- Self-modulate their QoS priority depending on how well their respective QoS requirements are met
- Use the integrated QoS regulators at ingress points to CMN-700

It is possible to include non-QoS-aware devices in the system, but still have these devices meet the QoS modulation requirement of the QoS architecture. To meet this requirement, CMN-700 includes inline regulators that perform the QoS functionality without the requesting device requiring any awareness of QoS. A *QoS Regulator* (QR) provides an interstitial layer between an RN and the interconnect. The QR monitors how the bandwidth and latency requirements of the RN are met, and does in-line replacement of the RN-provided QoS field. The QR adjusts the QoS field upwards for higher priority in the system and downwards for lower priority.

Component Aggregation Layer

A *Component Aggregation Layer* (CAL) allows multiple devices to connect to a single device port on an XP.

CMN-700 has multiple types of CAL. The different types of CAL support connection of different types of devices.

CAL2, CAL that can connect to two devices. CAL2 supports the following device types:

CALBYP2

- RN-F_CHIB_ESAM
- RN-F_CHIC_ESAM
- RN-F_CHID_ESAM
- RN-F_CHIE_ESAM
- RN-I
- RN-D
- HN-F
- HN-I
- HN-P
- SBSX
- MTSX
- SNF
- CXRH
- CCG

CAL4, CAL that can connect to four devices. CAL4 only supports the RN-F_CHID_ESAM device type.

CALBYP4

- RN-F_CHID_ESAM
- RN-F_CHIE_ESAM

The bypass variants of each CAL type, CALBYP2 and CALBYP4, are the same except for their flit buffering behavior. If there are enough credits for a flit to be sent between the device and MXP, CALBYP2 and CALBYP4 incur no latency when passing flits. However, using CALBYP2 or CALBYP4 increases timing pressure. CAL2 and CAL4, the variants of each CAL type without bypass functionality, always incur a single cycle of latency, even if there are credits to send flits. This behavior also applies to the MXP to device path.

All devices that are connected to a single CAL must be of the same type and you must configure them identically. The following figure shows a CAL2 example configuration.

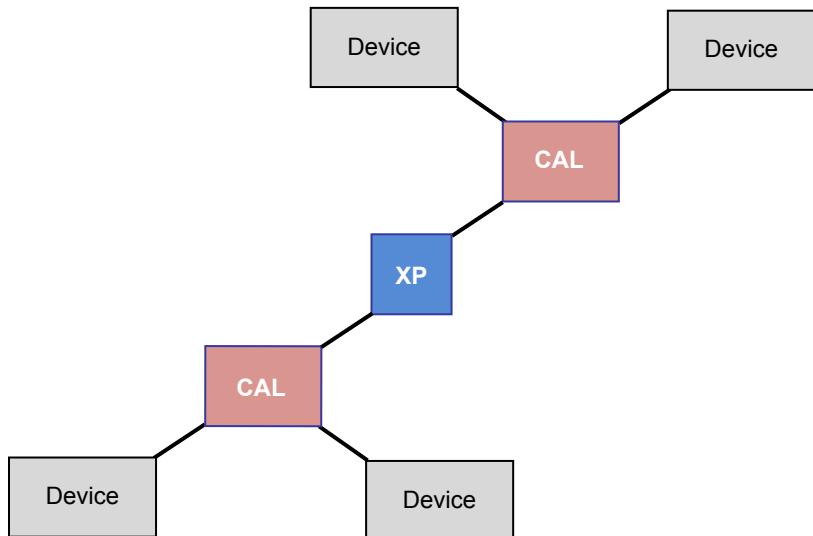


Figure 2-6 CAL2 example configuration

Credited Slices

You can configure various optional credited register slices in your CMN-700 system. These Credited Slices can help with timing closure.

Credited Slices enable synchronous but higher latency communication at any point in the system.

CMN-700 includes the following optional Credited Slices:

Mesh Credited Slice

Placed between XPs. For more information, see [Mesh Credited Slice on page 2-49](#).

Asynchronous Mesh Credited Slice

Placed between XPs that are in different clock domains. For more information, see [Asynchronous Mesh Credited Slice on page 2-49](#).

Device Credited Slice

Placed between a device and a CAL, or a device and an XP. For more information, see [Device Credited Slice on page 2-51](#).

CAL Credited Slice

Placed between a CAL and an XP. For more information, see [CAL Credited Slice on page 2-51](#).

The slices are simple repeater-flop structures that are applied across the entire communication boundary. The supported number of Credited Slices of each type is specified in [1.5.3 Device placement and configuration on page 1-27](#).

The following figure shows where various Credited Slices fit in the structure of the mesh. The example mesh includes two MCSs (denoted as MCSX and MCSY according to the X or Y direction of the link) and a DCS.

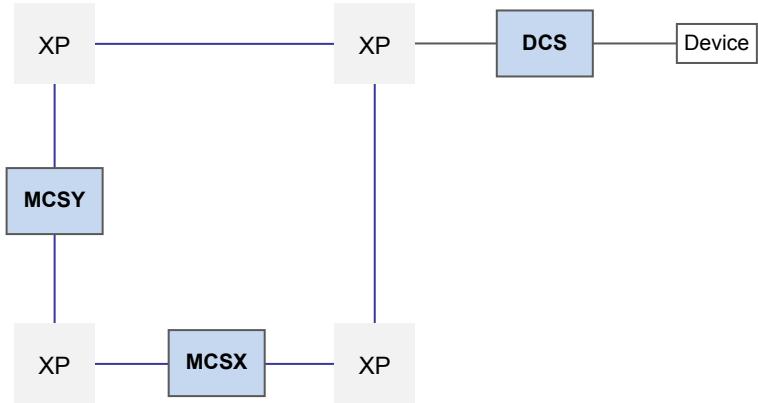


Figure 2-7 Example MSCX, MSCY, and DCS configuration

The following figure shows where various Credited Slices fit in the structure of the mesh. The example mesh includes two MCSs (denoted as MCSX and MCSY according to the X or Y direction of the link), a DCS, and a CCS.

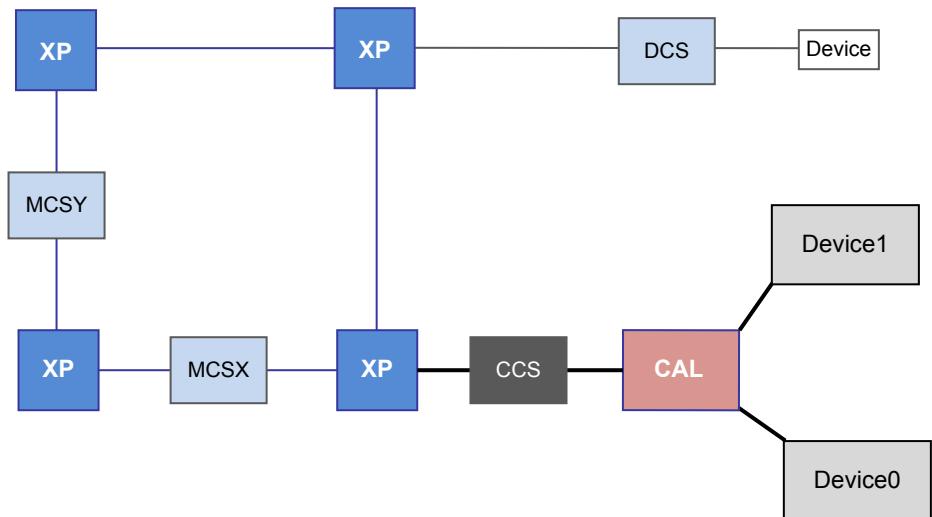


Figure 2-8 Example MSCX, MSCY, CCS, and DCS configuration

The following figure shows a full mesh and device topology with MCS and DCS.

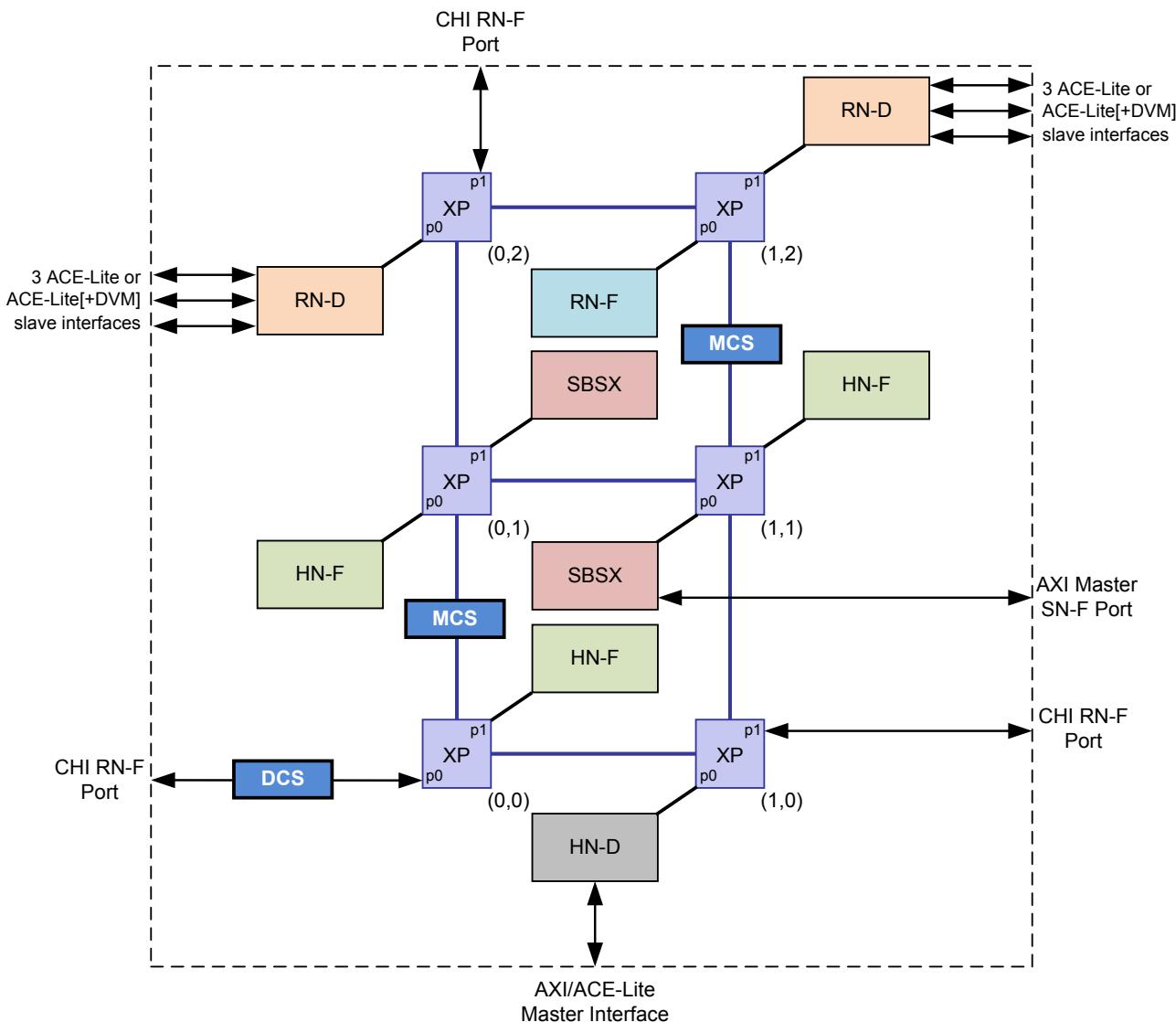


Figure 2-9 CMN-700 system with MCS and DCS

Mesh Credited Slice

You can configure one or more *Mesh Credited Slices* (MCSs) between CMN-700 XPs. MCSs are optional register slices that can help timing closure in a CMN-700 system.

The CMN-700 mesh can operate with a single cycle of latency between XPs. However, depending on the fabrication process and the distance between XPs, a single-cycle XP-XP connection might limit frequency. In this case, one or more MCSs can be added to lengthen the XP-XP links. Register slices add link transfer latency, but also allow certain CMN-700 implementations to run at higher frequencies.

Each MCS on an XP-XP link adds an extra cycle between XPs. One to four MCSs can be added to any link between XPs.

An MCS that is placed between adjacent XPs in the same row is called an MCSX. Similarly, an MCS that is placed between adjacent XPs in the same column is called an MCSY.

Asynchronous Mesh Credited Slice

CMN-700 supports multiple asynchronous clock domains across the mesh. *Asynchronous Mesh Credited Slices* (AMCSs) perform clock domain crossing between two asynchronous mesh clock domains.

To configure multiple clock domains in the mesh, you must also configure an AMCS on all XP-XP links that span different clock domains.

— Restriction —

The AMCS does not perform frequency scaling, it only synchronizes traffic between asynchronous clock domains. All CMN-700 clock domains must operate at the same frequency.

When using AMCSs to create multiple asynchronous clock domains, you must divide the mesh into four quadrants, each with a single clock domain. These quadrants must be rectangular and include one or more XPs.

For more information on CMN-700 support for asynchronous clock domains, see [2.2.1 Clock domain configurations on page 2-73](#).

You must configure one or two surrounding MCSs for each AMCS on a link. The following figure shows the supported AMCS plus MCS topologies:

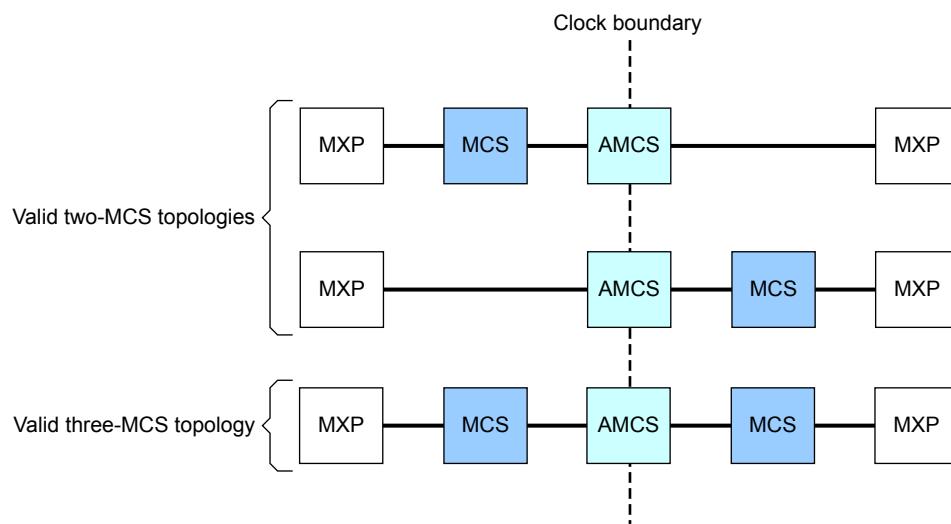


Figure 2-10 Valid AMCS topologies

The following figure shows an example CMN-700 topology with four asynchronous clock domains that are bridged by AMCSs. The clock domains must be arranged in the order that is shown in the following figure. Clock domain 0 must be the bottom-left quadrant.

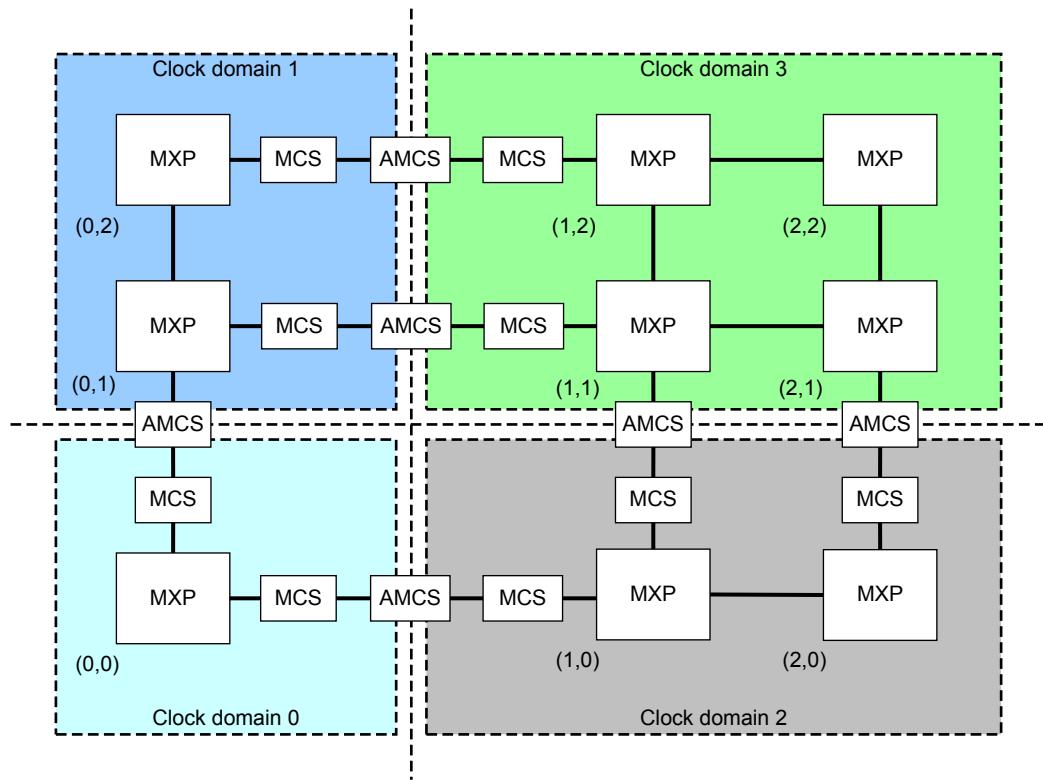


Figure 2-11 Example asynchronous mesh topology

Device Credited Slice

You can configure one or more *Device Credited Slices* (DCSs) on a link between a device and an XP. DCSs help with timing closure in a CMN-700 system.

DCSs are optional register slices that you can add to your CMN-700 configuration. You can add up to four DCSs on any link between a device and an XP.

CAL Credited Slice

You can configure up to two CCSs on the link between a CAL and an XP.

CHI Domain Bridge

The *CHI Domain Bridge* (CDB) bridges two CHI interfaces that operate in two different clock domains, power/voltage domains, or both.

For more information about the CDB, see the *Arm® Neoverse™ CMN-700 Coherent Mesh Network Configuration and Integration Manual*, which is only available to licensees.

AMBA® Domain Bridge

The *AMBA Domain Bridge* (ADB) bridges two AXI, ACE5-Lite, or ACE5-Lite-with-DVM interfaces that operate in two different clock domains, power/voltage domains, or both.

For more information about the ADB, see the *Arm® Neoverse™ CMN-700 Coherent Mesh Network Configuration and Integration Manual*, which is only available to licensees.

CXS Domain Bridge

The *CXS Domain Bridge* (CXSDB) bridges two CXS Issue A or B interfaces that operate in two different clock domains, power/voltage domains, or both.

For more information about the CXSDB, see the *Arm® Neoverse™ CMN-700 Coherent Mesh Network Configuration and Integration Manual*, which is only available to licensees.

Backward compatible RN-F support

CMN-700 is compliant with CHI-E, but can also contain RN-Fs that comply with CHI-D, CHI-B, and CHI-C. Certain restrictions apply to how CMN-700 handles transactions that are sent from older RN-Fs to maintain backwards compatibility.

The following table shows how all CMN-700 blocks handle specific backward compatible CHI-B, CHI-C, CHI-D, and CHI-E features.

Table 2-1 All blocks backward compatibility

All blocks	CHI-B	CHI-C	CHI-D	CHI-E
DBID[9:8]	Must be 0b00	Must be 0b00	Can be used	Can be used
DBID[11:10]	Must be 0b00	Must be 0b00	Must be 0b00	Can be used
SNP and DMT REQ TxnID[9:8]	Must be 0b00	Must be 0b00	Can be used	Can be used
SNP and DMT REQ TxnID[11:10]	Must be 0b00	Must be 0b00	Must be 0b00	Can be used

————— Note —————

If RN-F **TxnID[9:8] = 0b11**, DCT from CHI-B, CHI-C, or CHI-D RN-F to CHI-E RN-F cannot be done.

The following table shows how CMN-700 HN-Fs handle specific backward compatible CHI-B, CHI-C, CHI-D, and CHI-E features.

Table 2-2 HN-F backward compatibility

HN-F protocol	CHI-B	CHI-C	CHI-D	CHI-E
Requests from RN-F	Supported	Supported	Supported	Supported
DMT	Yes	Yes	Yes	Yes
DCT	Yes	Yes	Yes	Yes
Separate Response and Data	No	Yes	Yes	Yes
SnpPreferUnique	No	No	No	Yes
SnpQuery	No	No	No	Yes
New fields	MXP drives fixed values	MXP drives fixed values	MXP propagates new fields drives fixed values	MXP to propagate

The following table shows how CMN-700 DNs handle specific backward compatible CHI-B, CHI-C, CHI-D, and CHI-E features.

Table 2-3 DN backward compatibility

DN protocol	CHI-B	CHI-C	CHI-D	CHI-E
Requests from RN-F	Supported	Supported	Supported	Supported
CompDBID for DVM operations	No	No	Yes	No

————— **Note** —————

In a system with heterogeneous components, the system configuration must determine the lowest common DVM specification that is supported in the system. In such a system, set the enable_8_4_termination bit in the por_dn_cfg_ctl register to 1.

If enable_8_4_termination is set to 1, the DN detects Armv8.4-A DVM operations, suppresses their propagation, and responds in a protocol-compliant manner. This behavior avoids deadlocks and denial of service.

The DN does not send any error indication or log errors.

2.1.2 System configurations

CMN-700 can be configured to meet system requirements.

The following figure shows a 1×3 mesh for a small system configuration that contains single instances of RN-F, HN-F, RN-D, SN-F, and HN-D.

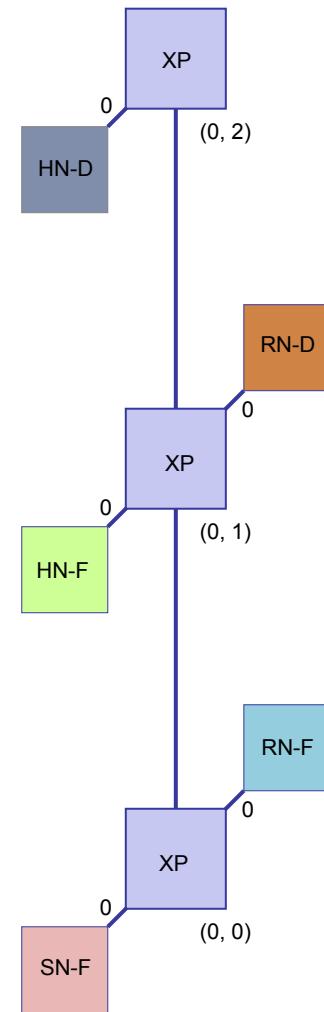


Figure 2-12 Mesh example 1×3

The following figure shows a 2×2 mesh for a medium system configuration with single and multiple instances of RN-F, HN-F, RN-D, SN-F, and HN-D.

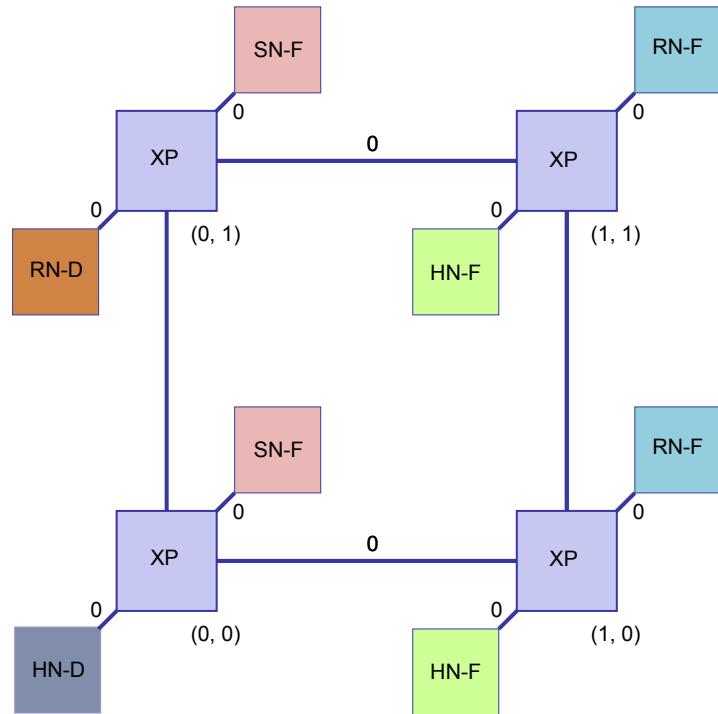


Figure 2-13 Mesh example 2 × 2

The following figure shows a 4×2 mesh for a medium system configuration with single and multiple instances of RN-F, HN-F, RN-D, SN-F, and HN-D.

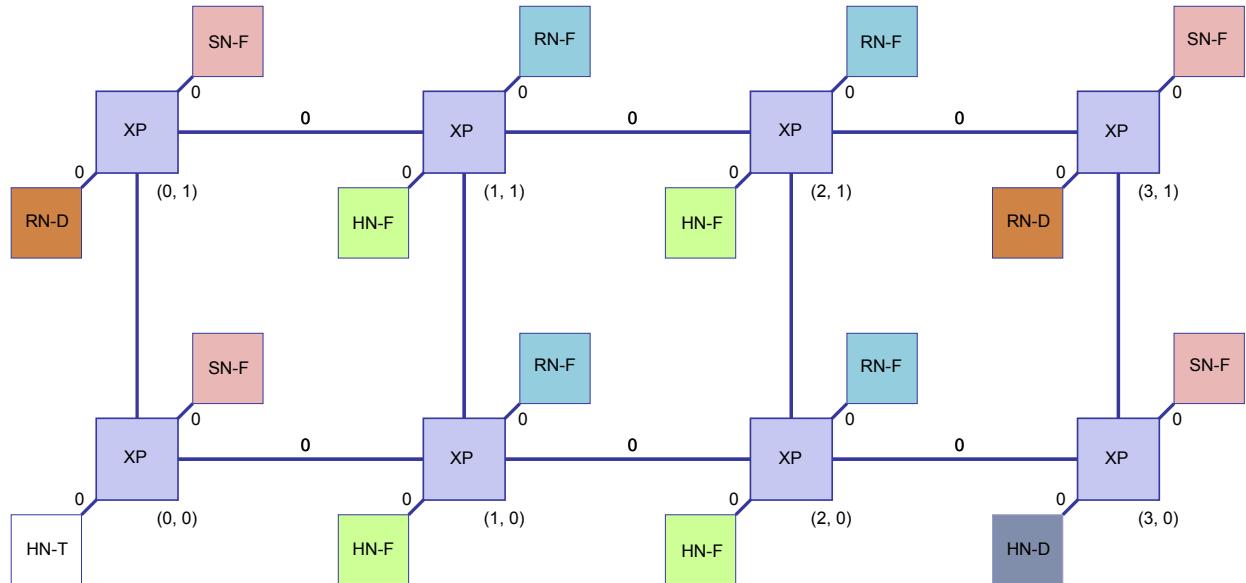


Figure 2-14 Mesh example 4 × 2

The following figure shows a 3×4 mesh for a large system configuration with multiple instances of RN-F, HN-F, RN-D, SN-F, and HN-D.

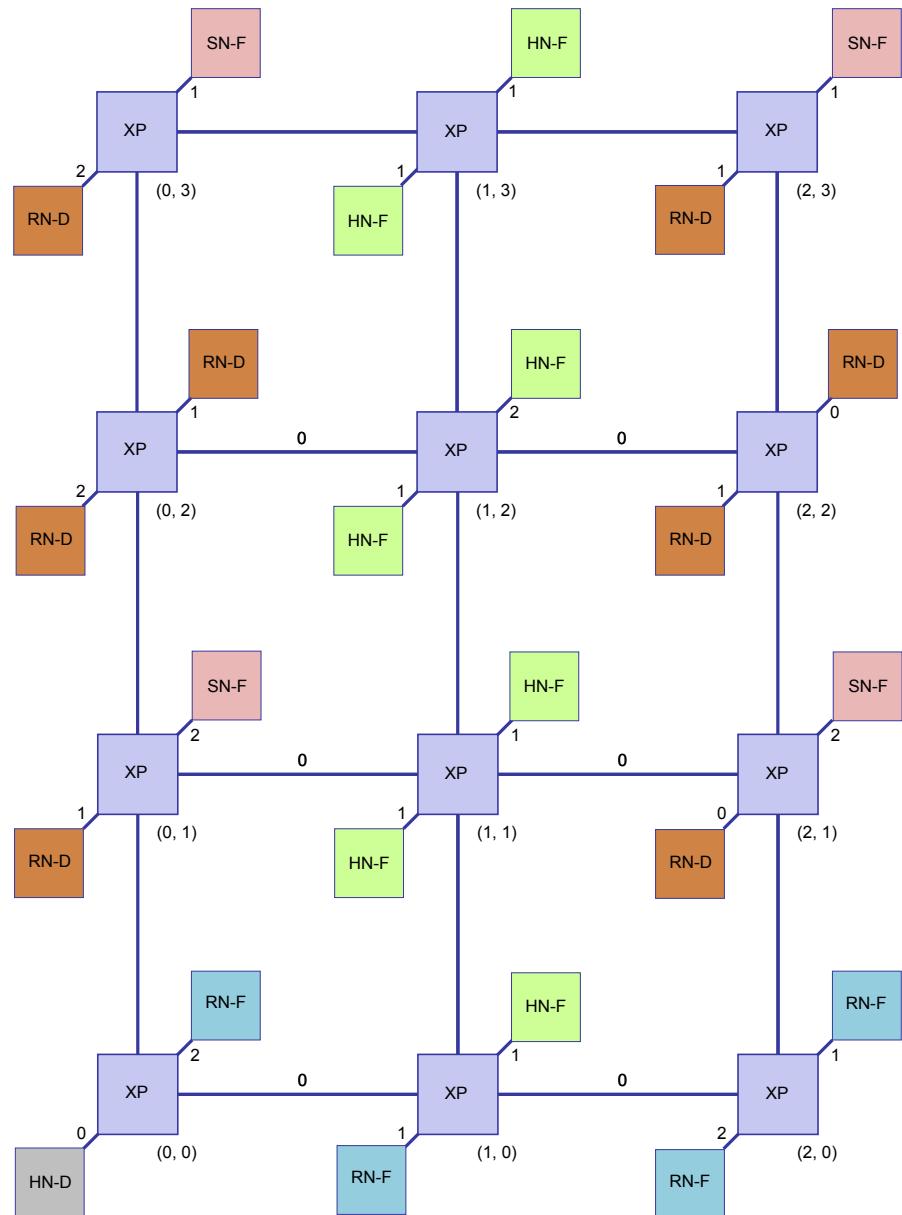


Figure 2-15 Mesh example 3×4

The following figure shows a 3×5 mesh for a large system configuration with multiple instances of RN-F, HN-F, RN-D, HN-I, SN-F, and HN-D.

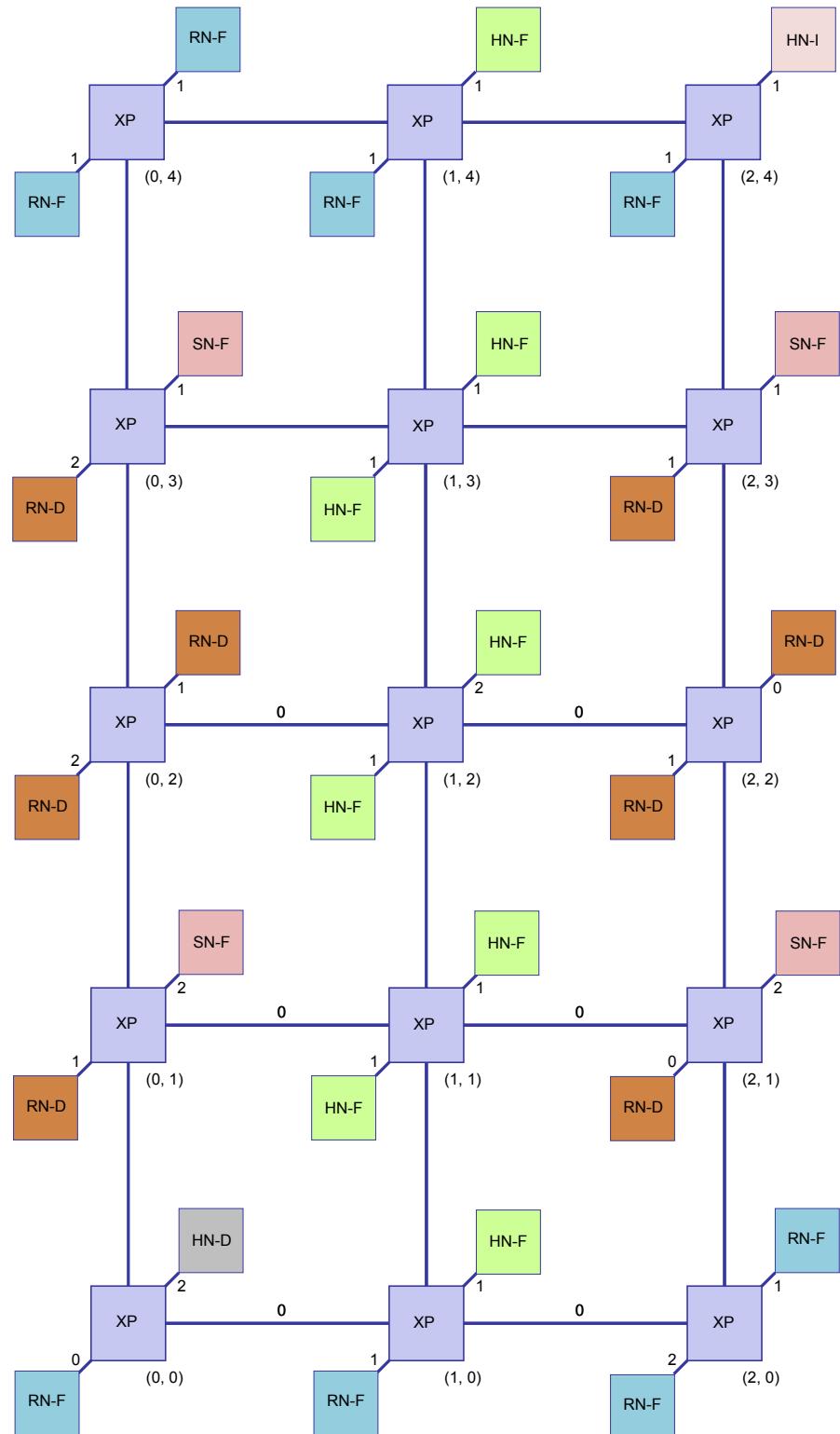


Figure 2-16 Mesh example 3 × 5

The following figure shows a 4 × 2 mesh for a medium system configuration with RN-F and HN-F CAL, shown by the gray areas.

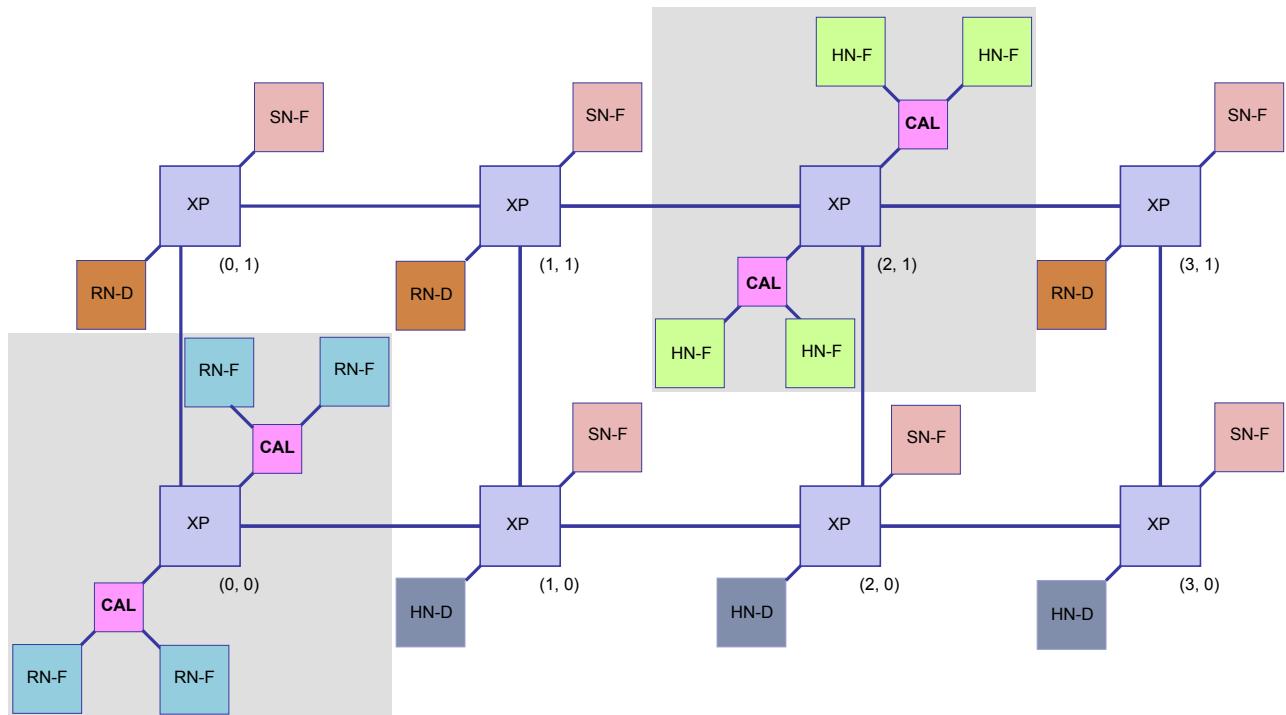


Figure 2-17 Mesh example 4×2 with CAL

2.1.3 CML system configurations

This section provides CML system configuration examples.

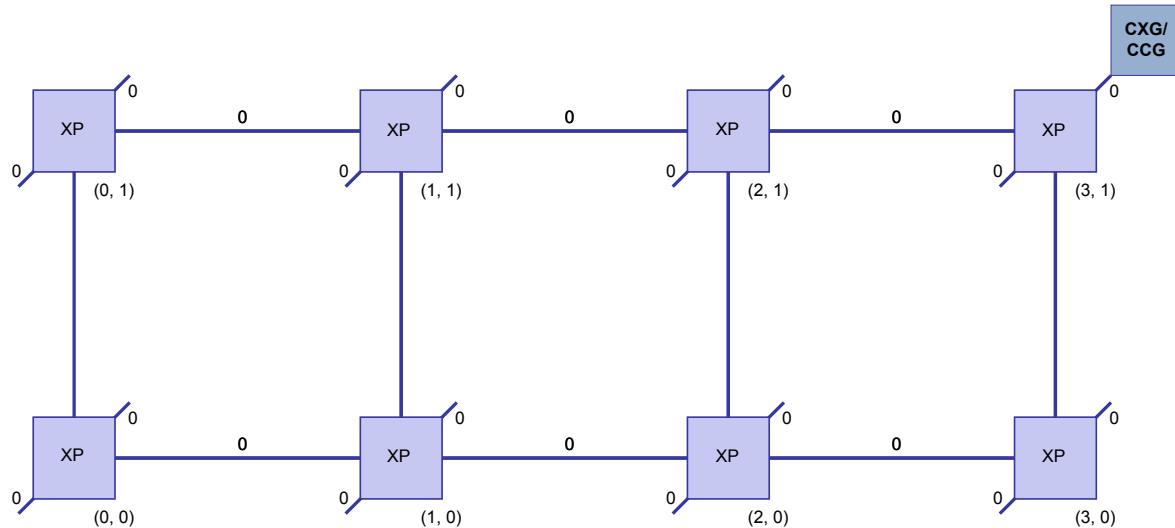


Figure 2-18 4×2 single CML mesh example

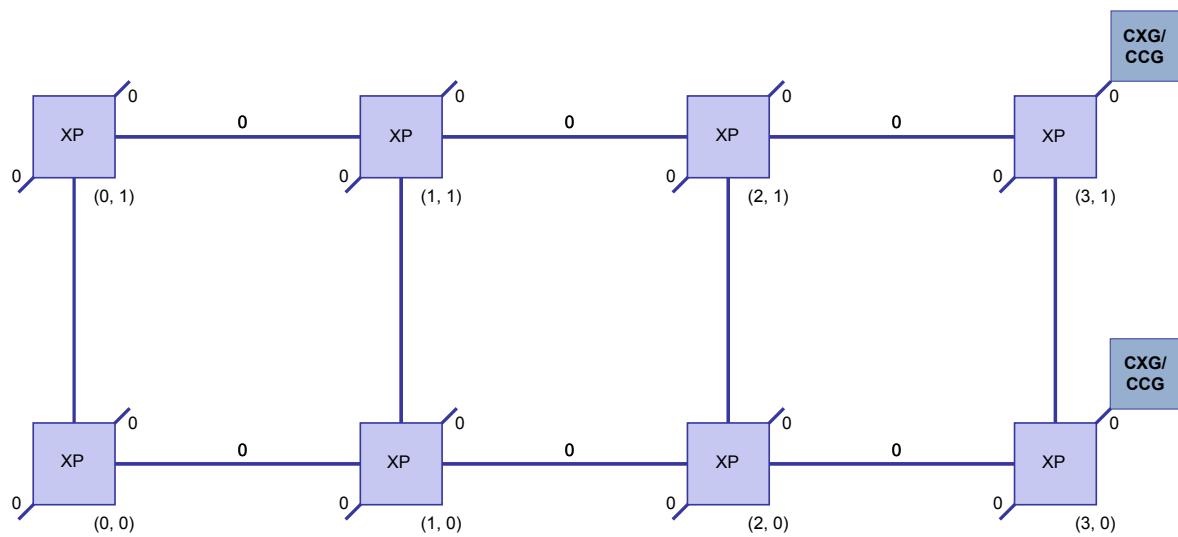


Figure 2-19 4×2 double CML mesh example

CXG/CCG components

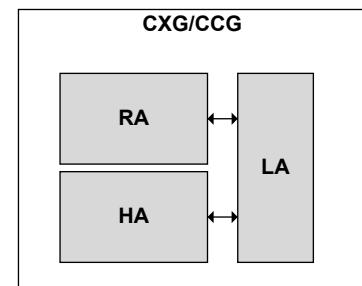
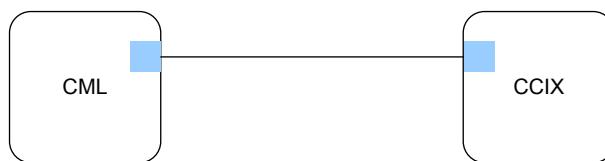


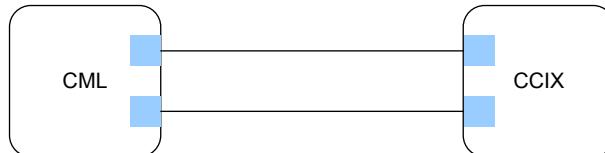
Figure 2-20 CXG/CCG block diagram with RA, HA, and LA

CCIX topologies

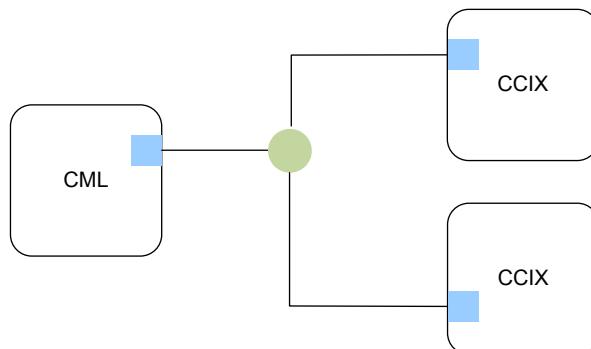
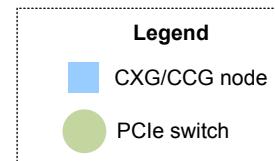
The following figure shows three simplified CCIX topology examples.



2-socket system with single CCIX port connection



2-socket system with Port Aggregation



3-socket system with external PCIe switch

Figure 2-21 CCIX topologies

————— Note —————

PCIe switch can only be used when connected through CXG node.

You can create more switched topologies by enabling port-to-port forwarding. For more information, see [2.4.15 CCIX1.1 port-to-port forwarding](#) on page 2-161.

CML Symmetric Multiprocessor support

A *Symmetric Multiprocessor* (SMP) allows for a shared, common OS and memory to operate on multiple chips.

CMN-700 supports SMP systems if the systems were built using the same version of CMN-700. For SMP systems, CCG block is needed to enable multi-chip SMP communications over 64B CXS (CML_SMP).

When set, the provided SMP mode option enables DVM, GIC-D, Exclusives, MPAM, CPU-Event, and selective CHI-D/E features across CML_SMP links using micro-architected mechanism.

Microarchitecture support for propagating Trace Tag across CML_SMP links is known as Remote Trace Tag. It is enabled in SMP mode only. Propagation of Remote Trace Tag is achieved by the sender CCG only on the outgoing CML_SMP request, and by the receiving CCG only from the incoming CML_SMP request. This feature ensures all subsequent CML_SMP messages that are part of the same transaction use the same CML_SMP TxnID. Similarly, the sender CCG only completes propagation of Remote Trace Tag on the outgoing CML_SMP Snoop, and the receiving CCG only from the incoming CML_SMP snoop.

For more information about programming CMN-700 for SMP mode, see [3.5 CML programming on page 3-1257](#).

CML Slave Agent support

CML Slave Agent mode is included to enable CCIX memory expansion and can be used to connect to CCIX Slave Agent Device. CXG block is required to connect to CCIX1.1 memory Slave Agent.

When configured in CCIX mode, CML supports CCIX-independent memory expansion where the CCIX link is used to communicate only with the Slave Agent on the remote chip.

For more information, see the *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*.

To enable this support, set the cxsa_mode_en bit in the [por_cxg_ccg_ra_cfg_ctl on page 3-336](#) register.

In this mode, CXRA/CCRA accepts SN bound CHI requests from HN-F and sends them to either CCIX Slave Agent or CXL Type3 memory device.

HN-F SAM is programmed according to target corresponding CXG/CCG unit for the remote address ranges of the expanded memory device. See HN-F SAM details in the [Chapter 3 Programmers model on page 3-214](#).

For further CXSA programming requirements, see [3.5 CML programming on page 3-1257](#).

CXHA passive buffer support

CMN-700 supports an optional passive buffer in CXHAs. This buffer lets the CXHA give extra CCIX Request and Data credits.

The CXHA passive buffer is present if the CXHA HA_PASS_BUFF_DEPTH parameter is set to a nonzero value.

CMN-700 uses RAMs to implement the data part of the passive buffer. This part of the passive buffer contains the following items:

- 64B of data
- Corresponding *Byte Enables* (BEs)
- Poison
- Any metadata that is associated with the data

CCIX CopyBack requests cannot be allocated in the passive buffer. Therefore, request and data credits from the CXHA active tracker are granted to service them. By default, CXHA reserves 64 CCIX Request and Data credits in total from the active tracker to service these CopyBack requests. However, you can configure this number to suit the requirements of your system, by programming the num_copyback_crds field of the por_cxg_ha_aux_ctl register. This field can take any value between the number of enabled SMP links to one less than the size of the write data buffer.

Note

The size of the write data buffer is equal to the value of the HA_NUM_WRBUF parameter.

When programming this value, you must ensure that enough active buffer entries are available to service the other non-CopyBack write requests.

If the passive buffer is present but any of the CCIX protocol links at the CXHA are configured to operate in non-SMP mode, then the passive buffer is disabled and bypassed. In this case, all CCIX request and data credits are granted from the CXHA active tracker. The disable_passive_buf field in the por_cxg_ha_aux_ctl register enables the passive buffer bypass mode during initial boot programming. For more information, see [por_cxg_ha_aux_ctl on page 3-587](#).

The CXRA tracks reserved CopyBack credits from the remote CXHA separately and uses them to send CopyBack requests. CXRA detects the number of reserved CopyBack credits during the initial link bring up process. Usually, this hardware detection mechanism is sufficient for a CXRA and CXHA pair to detect passive buffer mode and usage of special credits. However, if both of the following conditions are met, you can explicitly enable reserved credit usage during initial boot programming:

- It is known upfront that a CXHA has a passive buffer.
- All links of the CXHA are operating in SMP mode.

If these conditions are met, then software can use the lnk<X>_spcl_cbkwr_crd_en field in the por_cxg_ra_cxrptcl_link<X>_ctl register at boot. This field explicitly enables reserved credit usage.

CCG credit buffers

CMN-700 supports an optional passive buffer in CCGs. This buffer lets the CCG give extra CCIX Request and Data credits.

1. CCHA Request and Data credit buffers
2. CCLA link credit buffer
3. CCRA snoop credit buffer

CML credit requirements

Each CML port requires a minimum of one request and one data credit more than the total number of reservations. This number is enabled through the CXRA/CCRA configuration control register (por_cxg_ccg_ra_cfg_ctl).

This requirement applies to each enabled CCIX link at a given CCIX port. For example, by default all the reservations are enabled in the SMP mode. Therefore, a minimum of four request and four data credits must be granted per CCIX link. These credits are used by certain traffic types, such as QoS-15, to make progress in a loaded system. For more information, see the following registers: [por_cxg_ra_cfg_ctl on page 3-336](#) and [por_ccg_ra_cfg_ctl on page 3-810](#).

PCIe writes

To optimize PCIe write bandwidth, CCG supports high bandwidth write streaming from PCIe RN-I or RN-D. When high bandwidth write streaming is enabled, RA issues early completion for non-last portions of the write requests sent by RN-I or RN-D. This helps reduce tracker lifetime inside RN-I and RN-D.

CCG also provides support for write gathering for PCIe writes. To enable write gathering at the endpoint, CCG creates uniquely identifiable write streams coming from different RN-I and RN-D in the system: CCG sends out requestor's RAID to remote HA RN-I with the hashed AWID and AXI PortID information received from the issuing RN-I or RN-D. Remote HA RN-I then issues a request to HN-P with all the identifying information that can be used downstream of HN-P to gather write bursts.

For more information on RN-I and RN-D write burst cracking and write gathering flow refer to see [6.1.1 RN-I and RN-D write burst cracking on page 6-1325](#).

Note

If PCIe RNI can issue W+CMO opcodes to CCG, then the high bandwidth write streaming flow cannot be used and must be disabled.

High bandwidth PCIe Write streaming flow in CCG can be disabled by setting DIS_RNID_WARLY_WCOMP, DIS_RNID_TNL_RETRY_TRK, and DIS_SEP_PCIE_WR_CHAIN config bits in the por_ccg_ra_aux_ctl register.

High bandwidth PCIe Write streaming flow to CCG from RN-I and RN-D can be disabled by setting DIS_PCI_CXRA_WR_STREAM in por_rni_aux_ctl register.

To uniquely identify the originating RN-I or RN-D for write gathering, software must make sure that it assigns a unique RAID to each RN-I and RN-D source starting from RAID 0.

PCIe reads

PCIe reads RN-I or RN-D and issues them to the remote HA RN-I. The remote HA RN-I then either issues a read burst when the target is HN-P, or cracks the read bursts into 64B chunks when the target is DRAM.

————— Note —————

PCIe Read bursts cannot be issued to a CCG that does not have a RN-I instantiated inside HA.

To indicate that remote HA RN-I is present set the remote_rni_present field of the por_ccg_ra_aux_ctl register.

For more information on RN-I and RN-D support for preservation of AXI burst for PCIe read transactions through the Interconnect when the request is targeting HNP and PCI_CXRA node types, see [6.1.2 PCIe Read Burst Preservation through Interconnect on page 6-1325](#).

For more information on RN-I and RN-D support for AXID-based targetID selection for HN-P targets, see [6.1.3 RN-I/RN-D AXID Based target selection on page 6-1325](#).

Programming

1. PCIe RN-I Programming:
 - a. For every address region if the target_nide is CCG, program RN SAM non_hash_mem_region_reg#{index}.region#{index}_target_type register with PCI_CXRA target type
 - b. Set the pcie_mstr_present field of the por_ccla_rni_cfg_ctl register. This programming indicates that one or more PCIe masters are present upstream.
2. CCG RA programming:
 - a. Set remote_rni_present field of the por_ccg_ra_aux_ctl register. This programming indicates that the HA on remote chip has RN-I instantiated inside it to support PCIe read burst and write gathering features.
3. CCG HA_RNI
 - a. Set the pcie_mstr_present field of the por_ccla_rni_cfg_ctl register. This programming indicates that one or more PCIe masters are present upstream.
 - b. Follow the programming sequence outlined in 3.4.6 RN-I and HN-I PCIe programming sequence

2.1.4 Structural configuration and considerations

CMN-700 supports various options that you can use to further configure the structure of the interconnect. Some of these options have specific restrictions and requirements, and modify the behavior of the interconnect from the default behavior.

CMN-700 supports the following structural options:

- Dual REQ, DAT, RSP, and SNP channels
- Dedicated RN-I resource for traffic through AXI ports
- *CCIX Port Aggregation Groups* (CPAGs)
- Using extra device ports on MXPs
- Connection of devices with multiple interfaces

Dual DAT and RSP channels

To reduce congestion within the mesh, CMN-700 provides a configuration option to double the number of DAT and RSP channels. This feature is useful if you have heavy traffic sources that saturate specific routing paths.

If the POR_2XDATRSP_EN_PARAM configurable parameter is enabled, the number of DAT and RSP channels per MXP-MXP connection increases from one to two. This option has the following benefits:

- Increases device upload bandwidth
- Increases MXP-MXP bandwidth

The following figure represents the dual DAT and RSP channels in the mesh.

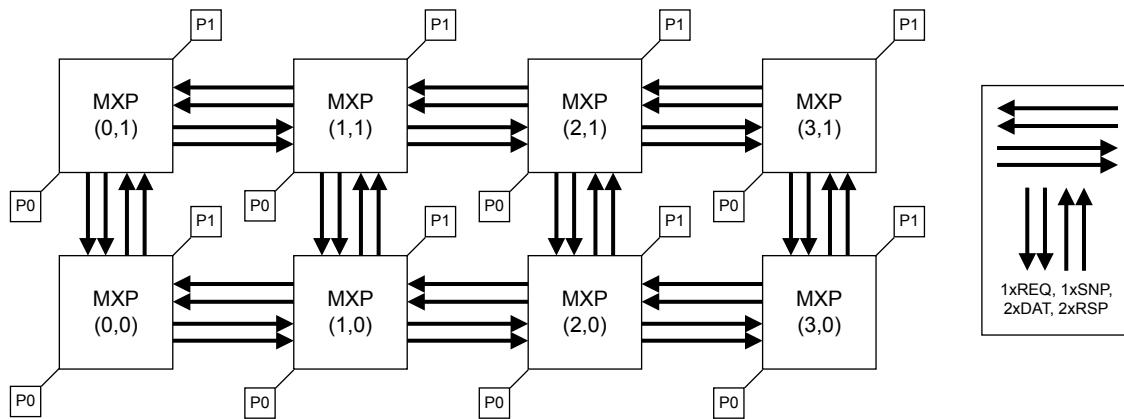


Figure 2-22 Dual DAT and RSP channel topology

Boot-programmable registers determine which of the dual channels to select at flit upload. For more information about configuring the selection, see [Dual DAT/RSP channel selection registers on page 2-66](#). For the procedure to program these registers, see [3.4.4 Program the dual DAT/RSP channel selection scheme on page 3-1243](#).

Dual DAT/RSP channel selection

If you enable the dual DAT/RSP channel feature, CMN-700 uses a TgtID-based selection scheme to determine which channel to upload DAT/RSP flits to. The interconnect also uses a static 2:1 multiplexer to download flits from the two channels.

The following figure shows the channel selection mechanism in the MXP. The figure shows the upload channel selection mechanism at port 0 and the download channel selection mechanism at port 1. However, port 0 can use the download channel selection mechanism and port 1 can use the upload channel selection mechanism. The mechanisms are similar for both DAT and RSP channels.

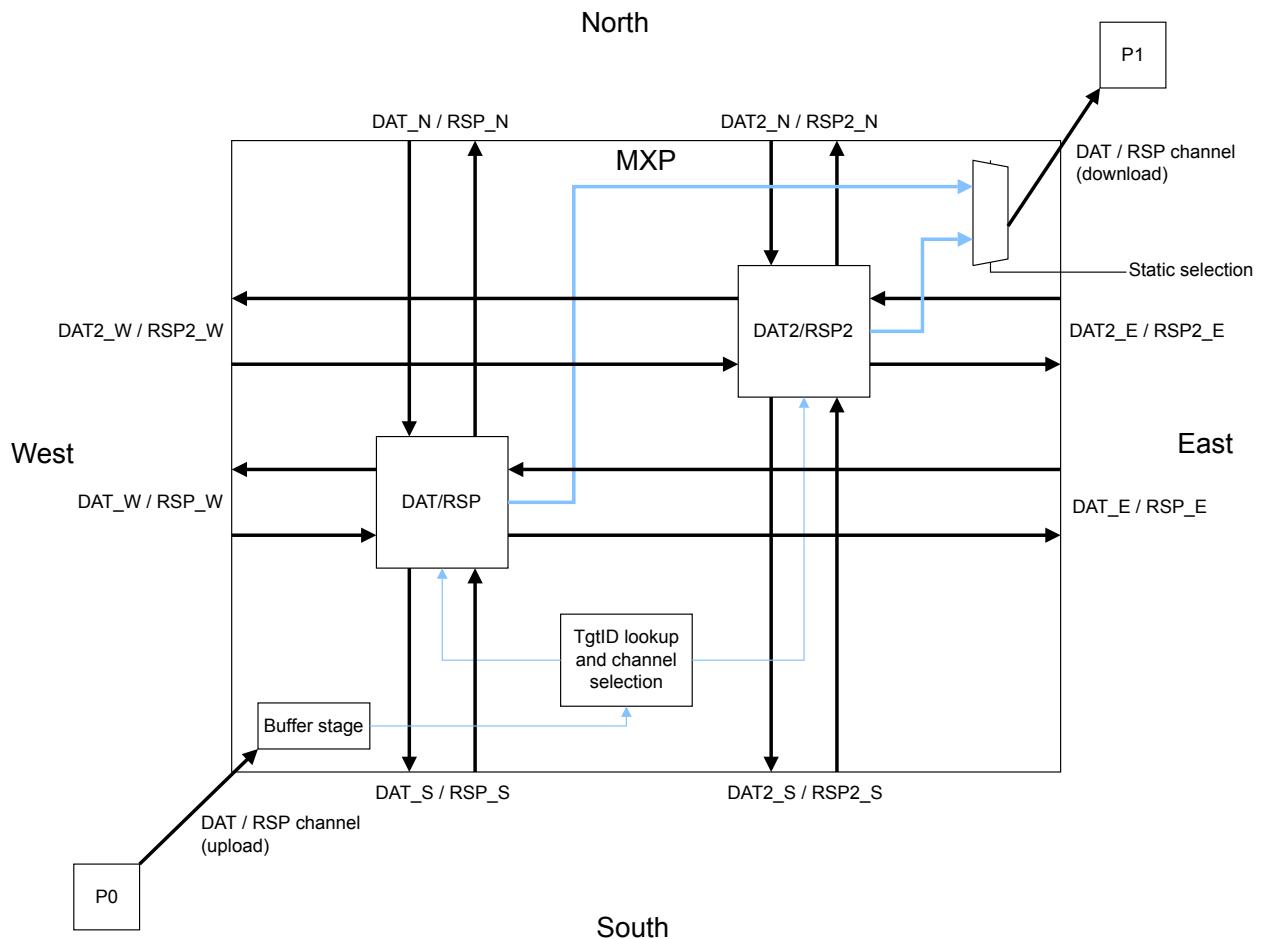


Figure 2-23 Dual DAT/RSP channel selection mechanism

Each mesh target is associated with one of the two DAT channels. All traffic to a single target is mapped to one of the two DAT channels. Likewise, all responses are mapped to one of the two RSP channels.

When you enable this feature by setting the configuration parameter, CMN-700 applies a default channel selection scheme. For mesh configurations where one of the mesh dimensions = 1 ($1 \times N$ configuration), traffic to all targets in the network use channel 0. Otherwise, traffic to targets on MXPs with an even XID uses channel 0 and traffic to targets on MXPs with an odd XID uses channel 1. This scheme is active until you specify a new scheme at boot by programming the channel selection behavior. For more information, see [3.4.4 Program the dual DAT/RSP channel selection scheme on page 3-1243](#).

————— Note ————

For the "POR_MESH_DEF_MESH_SEL_PARAM" value, the default scheme will be based on Even/Odd XID/YID. If the POR_MESH_DEF_MESH_SEL_PARAM value is:

- 0: Default mesh selection is based on Even/Odd target XID
- 1: Default mesh selection is based on Even/Odd target YID

After you have programmed the scheme, the MXP compares the TgtID of uploaded flits to the programmed *Lookup Table* (LUT). This LUT specifies which channel the MXP must send the flit to.

Enabling this feature also adds a static 2:1 multiplexer for each channel type. By programming this feature, you select one channel for the target to use. The MXP uses this multiplexer to ensure that only flits from the correct channel are downloaded to the device.

Dual DAT/RSP channel selection registers

Boot-programmable registers define the TgtID-based LUT. The MXP uses this LUT to select DAT/RSP channels for flits.

If the dual DAT/RSP channel feature is enabled, using `por_mxp_multi_mesh_chn_sel_*`, then each MXP contains 16 64-bit registers. The contents of these registers define the LUT that selects which channel a flit is assigned to. Each register can specify the assignment channel for four TgtIDs, so you can configure the behavior for up to 64 targets overall.

After the dual DAT/RSP registers have been programmed, any remaining unprogrammed TgtIDs are automatically assigned to channel 0. Unprogrammed registers and fields hold the reset value and are not part of the lookup. Programmed registers and fields are only valid for lookup when the `multi_mesh_chn_sel_reg_*_valid` bit of the register is set.

The following table shows the register format. Each 11-bit TgtID + CHN_SEL field specifies a TgtID and the DAT/RSP channel for flits with that TgtID.

Table 2-4 por_mxp_multi_mesh_chn_sel_N (N=0-15) register format

Bitfield	Description
[63]	VALID
[62:48]	2-bit CAL_DEV_CHN_MAP_SEL + 11-bit TgtID + CHN_SEL
[47:32]	2-bit CAL_DEV_CHN_MAP_SEL + 11-bit TgtID + CHN_SEL
[31:16]	2-bit CAL_DEV_CHN_MAP_SEL + 11-bit TgtID + CHN_SEL
[15:0]	2-bit CAL_DEV_CHN_MAP_SEL + 11-bit TgtID + CHN_SEL

Each 11-bit TgtID + CHN_SEL field has a reset value of `11'b0`. One or more fields can be left to hold the reset value in a programmed register with the `multi_mesh_chn_sel_reg_*_valid` bit set. In this case:

- Fields containing the reset value map TgtID 0 to channel 0 by default.
- If TgtID 0 is programmed to map to channel 1 by one of the register fields, then that value is used to determine the mapping. This mapping is used even if there are unprogrammed fields in the register.

The CAL_DEV_CHN_MAP_SEL (Channel Map select) for target devices behind CAL are associated with the corresponding TgtID fields:

Table 2-5 CAL_DEV_CHN_MAP_SEL corresponding TgtID fields

TgtID	CAL	Description
2'b00	CAL2	All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below (applicable for Mesh with > 2 device ports per XP) <ul style="list-style-type: none"> CAL2: DEV0,DEV1 are mapped to same channel,
2'b01	CAL4	All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below <ul style="list-style-type: none"> DEV0,DEV1,DEV2,DEV3 are mapped to same channel,
2'b10	reserved	-
2'b11	reserved	-

The MXP also contains a control register, `por_mxp_multi_mesh_chn_ctrl`. Program this register to indicate that the dual DAT/RSP channel selection configuration is complete.

The following table shows the control register format.

Table 2-6 por_mxp_multi_mesh_chn_ctrl register format

Bitfield	Description
[63:1]	Reserved
[0]	multi_mesh_chn_sel_programmed

For more information about programming these registers, see [3.4.4 Program the dual DAT/RSP channel selection scheme](#) on page 3-1243.

Dedicated RN-I resources for AXI port traffic

You can set up dedicated resources per AXI port in the RN-I and RN-D. Using this feature, you can ensure that one master that is connected to RN-I or RN-D does not block progress of traffic from another master.

For example, you can use this feature to connect SMMU and GIC components to RN-I or RN-D AXI ports, and ensure that real-time traffic from both components can progress.

To support this feature, the RN-I and RN-D have two sets of configuration registers per port and an auxiliary control register. You must program these registers to enable this feature.

Programming the registers for a port reserves a specific number of tokens for that port. Once these tokens are reserved, the master that is attached to that port is guaranteed an equivalent number of tracker entries and corresponding resources. Other masters that are connected to the RN-I or RN-D are blocked from using these reserved resources. Therefore traffic from the master with reserved tokens can progress. You can divide up RN-I or RN-D resources between different ports according to the relative resource needs of different masters.

The following table shows the configuration registers that you program to enable this feature. The table also shows the associated register fields and some properties of those fields.

Table 2-7 Configuration register fields for dedicated GIC and MMU RN-I resources

Register name	Register field	Bits	Reset value	Description
por_rni_aux_ctl	dis_port_token	[17]	1'b1	Enables and disables per port reservation for all ports for both read and write channels. Disables QoS15 reservation. ————— Note ——— CR_QPC_EN_Q enables QoS15 reservation.
por_rni_s0_port_control	s0_rd_token	[17:11]	7'h0	Number of reserved read tokens for port 0, per slice
por_rni_s0_port_control	s0_wr_token	[24:18]	7'h0	Number of reserved write tokens for port 0, per slice
por_rni_s1_port_control	s1_rd_token	[17:11]	7'h0	Number of reserved read tokens for port 1, per slice
por_rni_s1_port_control	s1_wr_token	[24:18]	7'h0	Number of reserved write tokens for port 1, per slice
por_rni_s2_port_control	s2_rd_token	[17:11]	7'h0	Number of reserved read tokens for port 2, per slice
por_rni_s2_port_control	s2_wr_token	[24:18]	7'h0	Number of reserved write tokens for port 2, per slice

For each port, software can program the number of tokens that correspond to the number of reserved entries in the tracker per slice. The token fields are 7 bits wide to accommodate the maximum tracker size of 96 entries. For a smaller tracker size, only the appropriate bit ranges are used. For example, for a 16-entry tracker, only bits [3:0] of the 7-bit field are used to indicate the number of reserved tokens.

When you enable this feature using the dis_port_token field of the por_rni_aux_ctl register, QoS15-based reservation is disabled.

The number of reserved entries per port is equal to the programmed token value plus one. For example:

If s0_rd_token = 3

Four tracker entries are reserved.

If s0_rd_token = 0

One tracker entry is reserved.

If the total number of the programmed reserved entries from all ports is larger than the number of read tracker entries in the slice minus 1, the number of tokens has been misprogrammed. To prevent overflow because of misprogramming, the RN-I reserves a specific number of tokens for each port. The RN-I uses a quarter of the programmed value of each port to calculate the number of reserved entries. If this value is a fraction, then the remainder is rounded to 0.

Example 2-1 Programmed values for eight-entry tracker

Consider the following programmed values for a tracker with eight entries:

s0_port_token = 4

Reserving five entries.

s1_port_token = 0

Reserving one entry.

s2_port_token = 1

Reserving two entries.

In this case, the total number of reserved entries is eight, and the tracker has no unreserved entries. However, the number of tracker entries minus one is seven, and so the registers are misprogrammed. In this case, the RN-I divides each register value by four to calculate the final number of reserved entries:

s0 = 4/4 = 1

Two entries are reserved.

s1 = 0/4 = 0

One entry is reserved.

s2 = 1/2 = 0

One entry is reserved.

Therefore, a total of four entries are reserved for this configuration. The tracker has four unreserved entries.

CCIX Port Aggregation Groups

The CMN-700 CML configuration supports up to 32 CXG/CCGs. These CXG/CCGs can be grouped in up to 16 *CCIX Port Aggregation Groups* (CPAGs).

This feature can be used when connecting two or more chips together with multiple ports between the chips. For example, the following figure shows three chips that are connected by four CXGs that are grouped into two CPAGs.

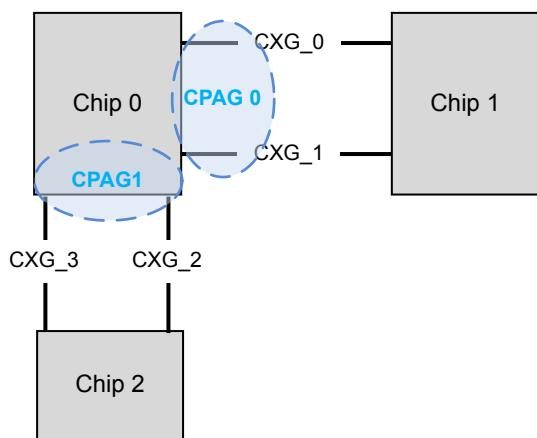


Figure 2-24 CCIX Port Aggregation Groups

In the preceding example, the following CPAGs are present:

- CPAG_0** CPAG with two CCIX ports (CXG_0 and CXG_1) to connect Chip 1 to Chip 0.
- CPAG_1** CPAG with two CCIX ports (CXG_2 and CXG_3) to connect Chip 2 to Chip 0.

To enable CPAG, both RN SAM and HN-F registers must be programmed accordingly in each chip.

Support for extra device ports on MXPs

You can expand the number of device ports on an MXP beyond the default number of two. This feature reduces the power consumption of CMN-700 configurations by reducing mesh transfer activity between MXPs.

By default, a single MXP has two device ports and four mesh ports. CMN-700 also supports extra device ports on MXPs. The maximum permitted number of device ports per MXP is configuration-dependent. For example, a single-MXP configuration is possible where there are up to six device ports on the MXP.

Enabling extra device ports affects the following aspects of CMN-700:

- The overall system topology. For more information, see [Topology considerations when using extra device ports on page 2-69](#).
- The number and mapping of DTMs in the MXP. For more information, see [DTM changes when using extra device ports on page 2-71](#).
- The mapping scheme for node IDs. For more information, see [2.4.2 Node ID mapping for configurations with extra device ports on page 2-99](#).

Topology considerations when using extra device ports

The number of devices in your CMN-700 design determines the number of MXPs and device ports you need. The design is also affected by whether you choose to put a CAL on any of the device ports.

If you have more than 6-8 devices in your CMN-700 configuration, you must use a mesh configuration with multiple MXPs.

The following figure shows an example mesh configuration using extra device ports.

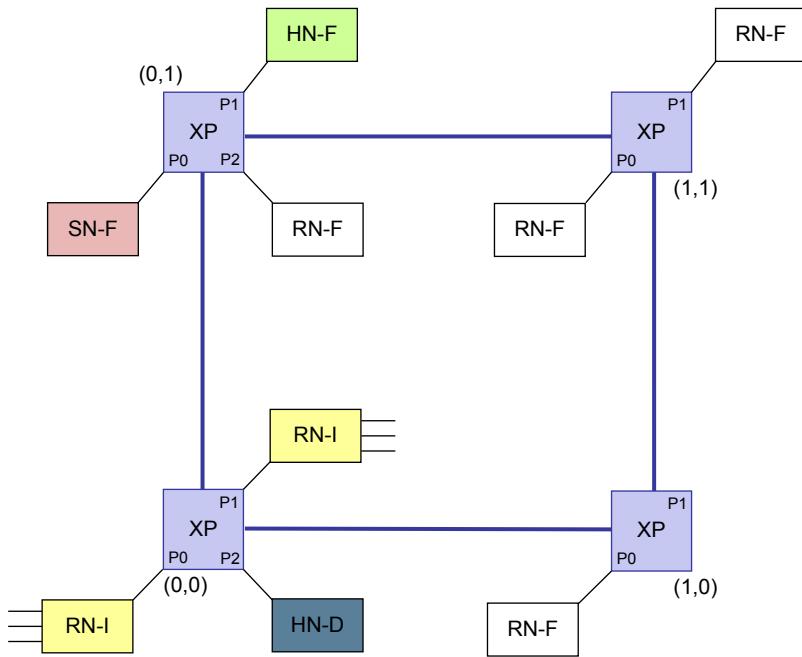


Figure 2-25 Mesh configuration using extra device ports

If you have six or fewer devices in your configuration, or eight or fewer if you are using CAL, you can use a single-MXP configuration. All devices can then be attached to one MXP, as shown in the following figure.

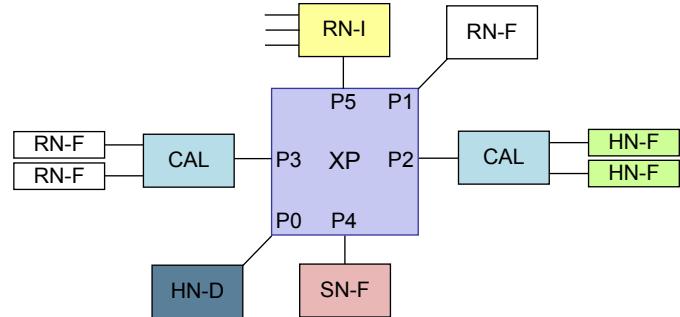


Figure 2-26 Single-MXP configuration with extra device ports

The following table shows the MXP design restrictions depending on whether you use a single-MXP or multiple-MXP configuration.

Table 2-8 MXP design restrictions when using extra device ports

MXP configuration type	Maximum number of device ports per MXP	Maximum number of devices per MXP	Maximum number of supported ESAMs per MXP
Single MXP	6	8	4
MXP in mesh configuration without HN-D or HN-T connected	4	8	4
MXP in mesh configuration with HN-D or HN-T connected	3	5, including HN-D or HN-T	4

A mesh configuration is a configuration with more than one MXP, in other words, not a single-MXP configuration. If the MXP in a mesh configuration has more than two device ports, HN-D or HN-T is only permitted on device port P2.

You can only attach a maximum of four RN-Fs to an MXP directly. This restriction is due to the maximum supported number of ESAMs per MXP, which is four for all configuration types. To increase the number of RN-Fs on an MXP, you must use a CAL.

DTM changes when using extra device ports

A configuration parameter determines whether DTMs are replicated in MXPs to support extra device ports. Each port is mapped to a specific DTM and both the configuration parameter value and the overall system configuration determines how the ports are mapped.

Note

For a mesh configuration, up to four device ports are permitted per MXP. For a single-MXP configuration, up to six device ports are permitted on the MXP. For more information, see [Topology considerations when using extra device ports on page 2-69](#).

To enable support for multiple DTMs in MXPs, set the `MXP_MULTIPLE_DTM_EN` parameter = 1. If this parameter is set to 1, then the DTM in each MXP is replicated according to the number of device ports on the MXP. Each DTM supports up to two device ports. The different interconnect configurations use the following DTM mappings:

Mesh configuration

- DTM0 supports P0 and P1.
- DTM1 supports P2 and P3.

Single-MXP configuration

- DTM0 supports P0 and P1.
- DTM1 supports P2 and P3.
- DTM2 supports P4 and P5.

If the `MXP_MULTIPLE_DTM_EN` parameter = 0, each MXP has a single DTM. In this case, each DTM supports the following device ports:

Mesh configuration

Single DTM per MXP supports P0, P1, P2, and P3.

Single-MXP configuration

Single DTM per MXP supports P0, P1, P2, P3, P4, and P5.

————— Note ————

If a single DTM supports more than two ports, it might not be possible to monitor all the events on different ports at the same time. However, using a single DTM saves on the implementation area.

2.1.5 DSU and DMC AXI Utility Bus

CMN-700 will provide a mechanism to interface with an AXI5 Utility Bus, which is a simplified version of AXI5. The accesses will be either 32b or 64b in size, and single data transfer with no burst. The utility bus will be connected to RN-F and SN-F node locations as an AXI5 master port.

The utility bus ports are identified with a logical ID. One set of logical IDs are contiguously numbered for RN-F; and another set for SN-F. To send a request to those ports, software should look up the logical ID base information from `por_mxp_p<0..5>_info.dslogicalid_base_p<0..5>` and `por_mxp_p<0..5>_info.dmc_logicalid_base_p<0..5>`. And there can be 0, 1, or 2 AXI Utility bus ports per device. This is identified by `por_mxp_p<0..5>_info.dsu_num_p<0..5>` and `por_mxp_p<0..5>_info.dmc_num_p<0..5>`.

The DSU address region starts from DSU_PERIPHBASE. Each DSU occupies a fixed 1MB space. Up to 256 DSU can be supported. The total DSU address space will be the total number of DSU rounded up to next power of 2. And a base aligned with this address space should be driven to DSU_PERIPHBASE. The DSU address is composed of [LSB of DSU_PERIPHBASE -1 : 20] [19:0]. The lower 20 bits is DSU offset, and bits above is logical ID.

The DMC address region starts from DMC_PERIPHBASE. Each DMC occupies a fixed 16MB space. Up to 16 DMC can be supported. The total DMC address space will be the total number of DMC multiplied by 16. And a base aligned with this address space should be driven to DMC_PERIPHBASE. The DMC address is composed of [LSB of DMC_PERIPHBASE -1 : 24] [23:0]. The lower 24 bits is DMC offset, and bits above is logical ID.

All three bases, CFGM_PERIPHBASE, DSU_PERIPHBASE, and DMC_PERIPHBASE are recommended to reside within the same lower 4GB aligned space, this is to facilitate APB access. It is required that bits [31:28] are different between CFGM_PERIPHBASE and DSU_PERIPHBASE/DMC_PERIPHBASE.

————— Note ————

BRESPU and RRESPU from AXI5 Utility bus will be propagated back on APB PSLVERR; but they will be dropped on CHI response.

2.2 Clocks and resets

CMN-700 provides a hierarchical clocking microarchitecture which enables dynamic clock management for power efficiency. It also has a global reset signal.

CMN-700 has a *High-level Clock Gating* (HCG) mechanism for clock signal management during periods of inactivity. For more information, see [2.2.5 High-level Clock Gating on page 2-79](#).

This section contains the following subsections:

- [2.2.1 Clock domain configurations on page 2-73](#).
- [2.2.2 CXG clock inputs on page 2-75](#).
- [2.2.3 Clock hierarchy on page 2-77](#).
- [2.2.4 Clock enable inputs on page 2-78](#).
- [2.2.5 High-level Clock Gating on page 2-79](#).
- [2.2.6 External Clock Controller on page 2-79](#).
- [2.2.7 CXG clock management on page 2-80](#).
- [2.2.8 Reset on page 2-81](#).
- [2.2.9 CXG reset on page 2-81](#).

2.2.1 Clock domain configurations

CMN-700 supports either one synchronous or multiple asynchronous clock domains.

The following figure shows a CMN-700 configuration operating in a single and fully synchronous clock domain.

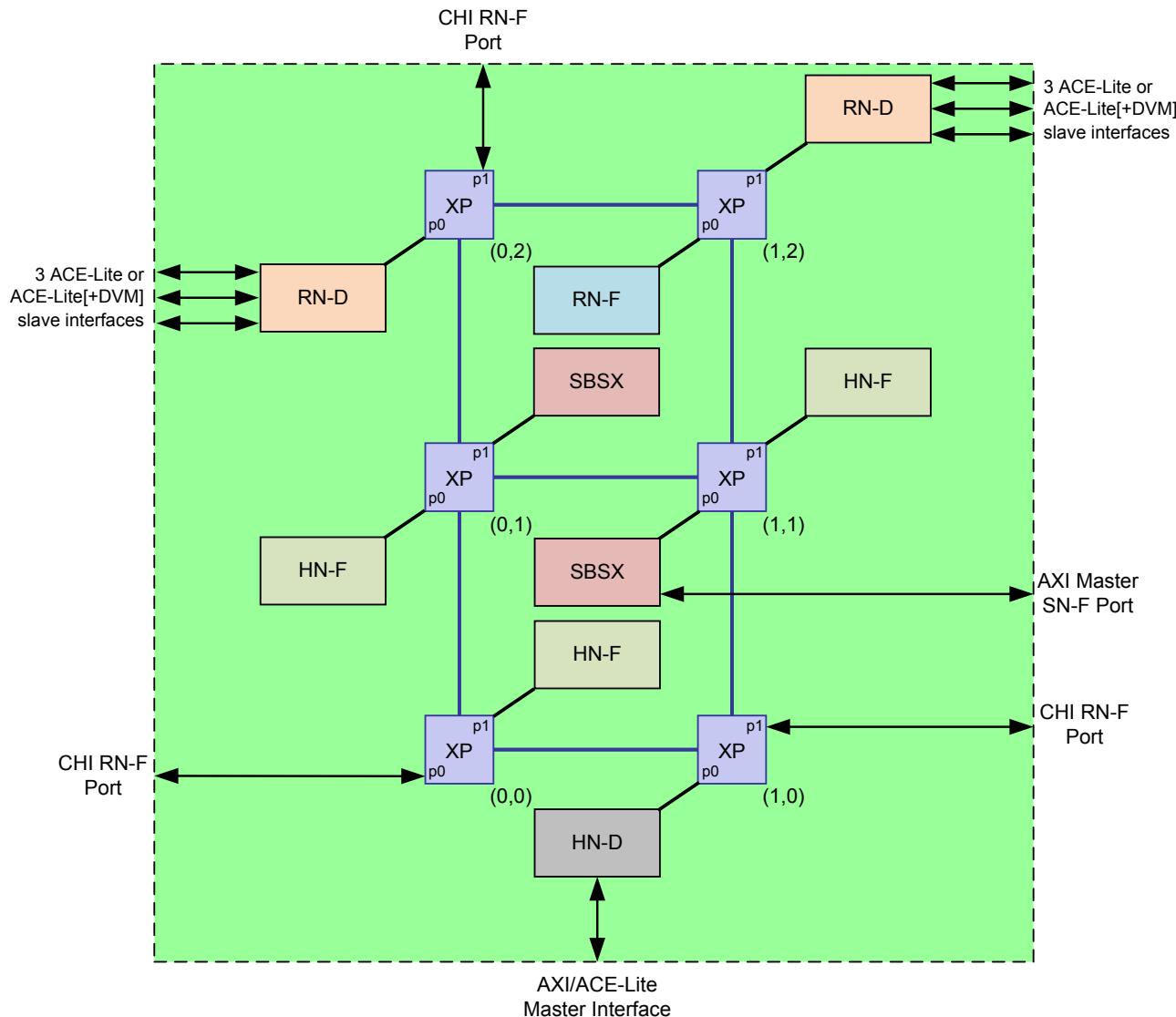


Figure 2-27 CMN-700 topology with fully synchronous clock domain

The global clock signal in a fully synchronous clock domain configuration is known as **GCLK0**.

For larger mesh topologies, synchronization and clock skew can be problematic because of the large distances that clock signals travel. Therefore, CMN-700 also supports dividing the mesh into four asynchronous clock domains. This feature is configured using Socrates.

The following figure shows an example CMN-700 configuration with four asynchronous clock domains.

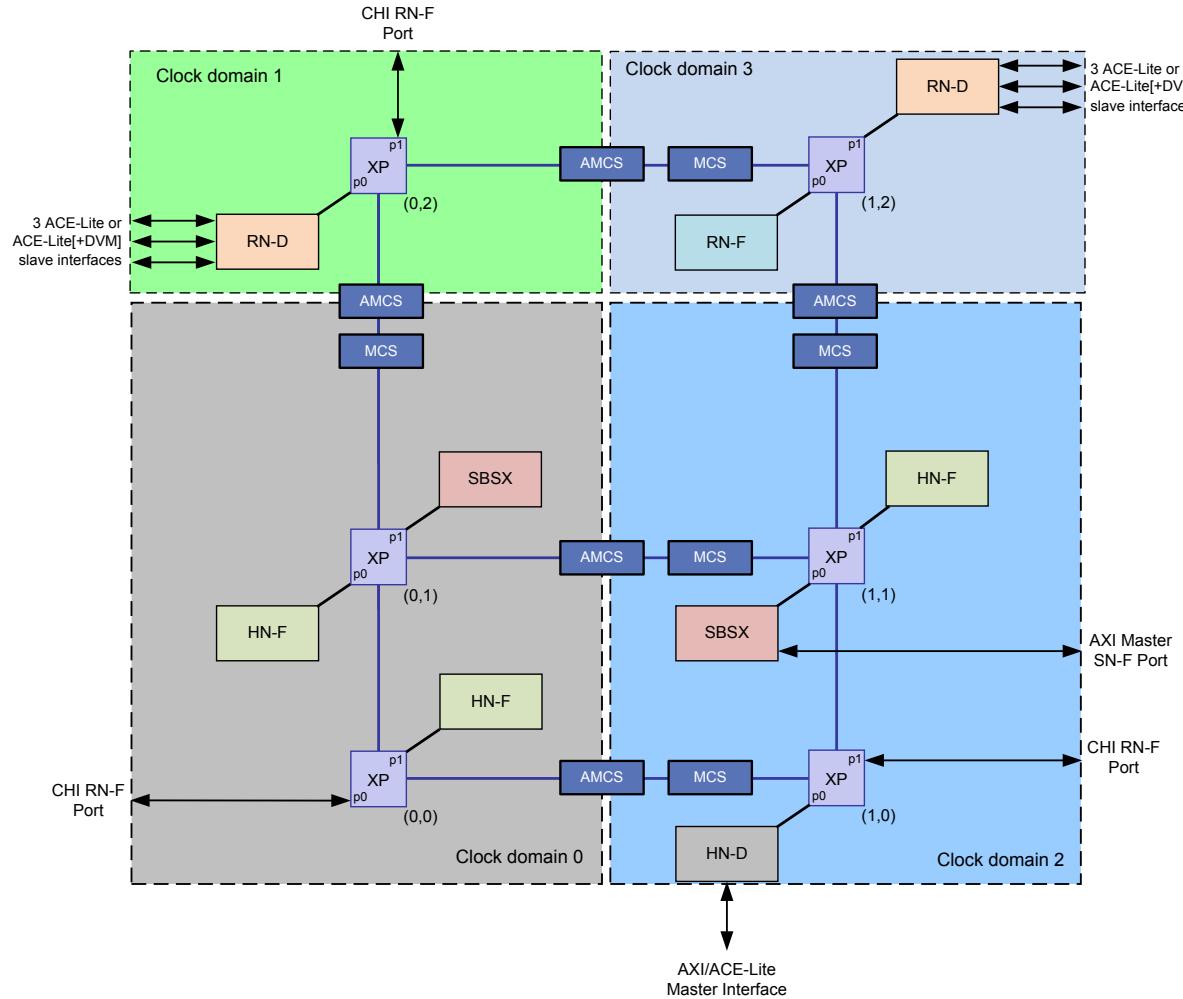


Figure 2-28 CMN-700 topology with two asynchronous clock domains

Each asynchronous clock domain is supplied by an individual clock input. These clock signals are known as **GCLK0**, **GCLK1**, **GCLK2**, and **GCLK3**, and collectively referred to as **GCLKn**.

If you configure the CMN-700 mesh to use multiple asynchronous clock domains, you must comply with the following restrictions:

- AMCSs must bridge asynchronous clock domains. Therefore, you must place AMCSs on the XP-XP links that span clock domains.
- You must configure the AMCS that bridges the two clock domains next to one or two MCSs in a chain. For more information about the allowed topologies, see [Asynchronous Mesh Credited Slice on page 2-49](#).
- The individual clock signals that supply each clock domain must run at the same frequency as each other, although they can be asynchronous.
- CMN-700 only supports rectangular clock domains containing one or more XPs. L-shaped or other clock domain topologies are not supported.
- 2x2 and smaller mesh configurations do not support multiple clock domains.
- A single Q-Channel controls all the clock signals. Therefore the clock domains cannot be gated separately, and must be powered up and down in the same way as a single synchronous clock domain.
- The mesh can only have a single power domain.

2.2.2 CXG clock inputs

There are two extra clock inputs for the CML/CXG configuration: **CLK_CGL** and **CLK_CXS**.

CLK_CGL is a copy of the CMN clock input used by the CXRH node associated with the CXLA, **GCLKn**. A separate clock input is provided to allow gating of the CGL clock domain independent of the **GCLKn** domain. **CLK_CXS** clocks the CXS interface logic, and can be synchronous or asynchronous to **GCLKn**. **CLK_CXS** can be driven with **CLK_CGL** for synchronous configurations, as the following figure shows.

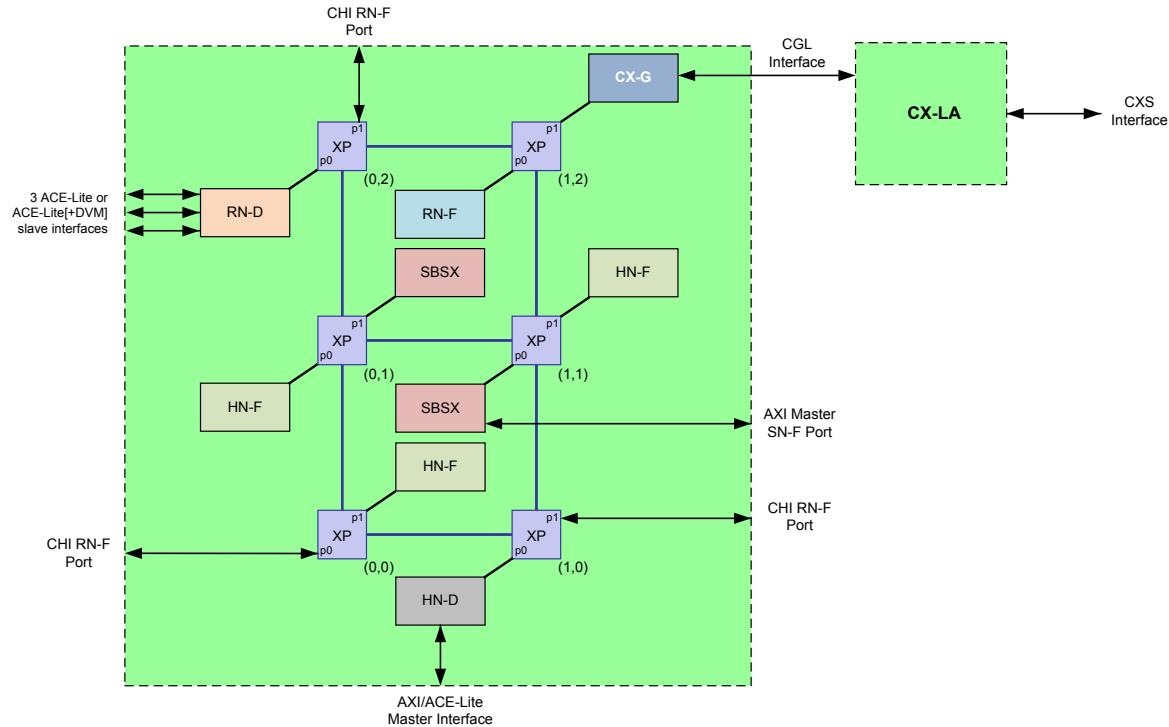


Figure 2-29 CMN-700 clock domains with synchronous CXS domain

————— Note —————

If the CXS interface is asynchronous to **GCLKn**, **GCLKn** clocks the corresponding CXRH logic. In other words the global clock signal, for the clock domain it sits in, clocks the CXRH.

The CXLA block contains an asynchronous domain bridge for configurations where the **CLK_CXS** domain is asynchronous to the **CLK_CGL** domain, as the following figure shows.

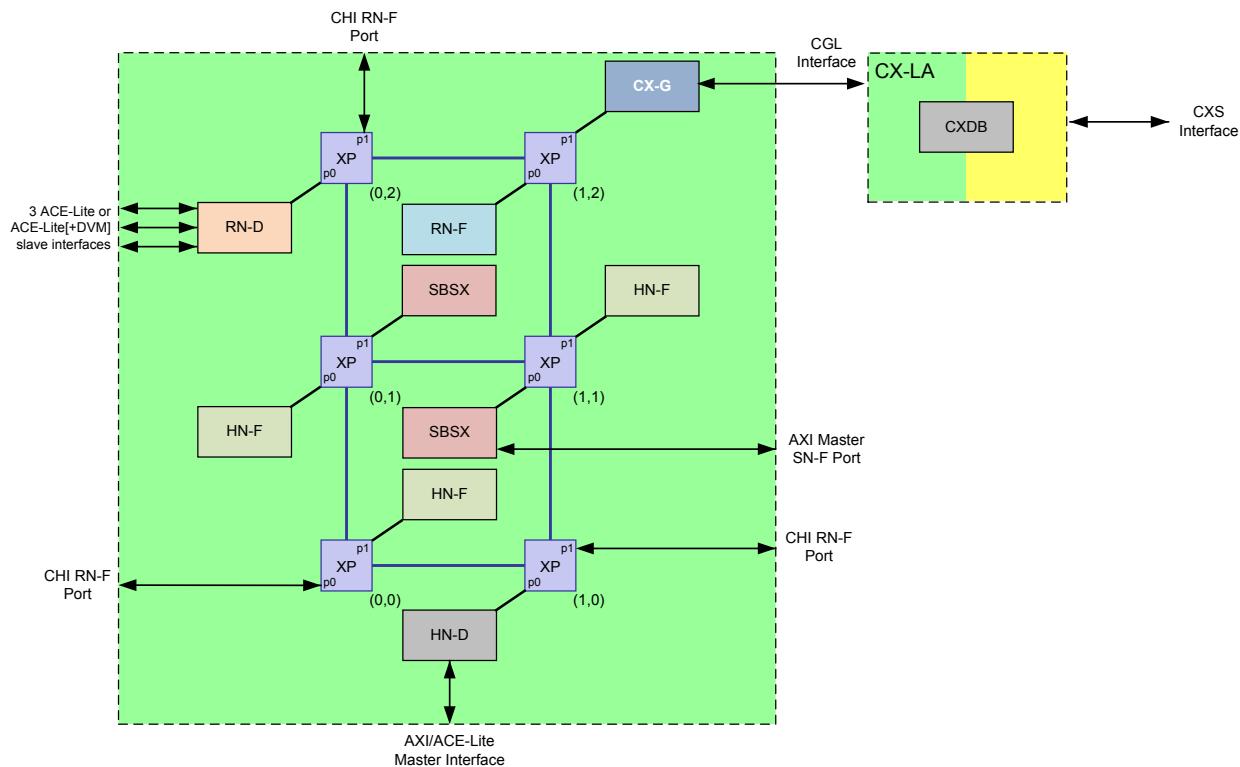


Figure 2-30 CMN-700 clock domains with asynchronous CXS domain

GCLKn clocks entire CCG logic including its CXS interface. Optionally, CXS Domain Bridge (CXSDDB) can be added on the CXS interface, if the external controller IP is running asynchronously to CMN clock (GCLKn).

2.2.3 Clock hierarchy

The clocking delivery and clock gating architecture are hierarchical.

Within the clock gating hierarchy, three levels of clocks are defined:

Global clock The global clock is the clock input to the CMN-700 system. Another level of clock gating or clock control outside of the system is likely to control the global clock that is provided by the SoC. Although it is not a system requirement, CMN-700 includes support for external clock control.

—————
Note—————

If you configure CMN-700 to use multiple asynchronous clock domains, a single Q-Channel controls each individual clock signal. Therefore, the individual clock domains cannot be separately gated. For more information, see [2.2.1 Clock domain configurations on page 2-73](#).

Regional clocks Regional clocks are created as an output of regional clock gaters that include a coarse enable for coarse-grained clock gating under idle or mostly idle conditions. Regional clock gaters can shutdown the clock network between regional and local gaters. Therefore, this level of hierarchy enables greater power reduction than is possible using local clock gating. The regional clock gaters are instantiated in and controlled by the CMN-700 RTL. The exact set of regional clocks is internal to CMN-700 and is not described in this book.

- Local clocks**
- Local clocks are created according to the following hierarchy:
 1. RTL creates fine grained enable signals.
 2. Fine grained enable signals control local clock gaters.
 3. Local clock gaters output local clock signals.

Local clock signals are used to directly clock sequential elements in CMN-700. The exact set of local clocks is internal to CMN-700 and is not described in this book.

The following figure shows the clocking hierarchy.

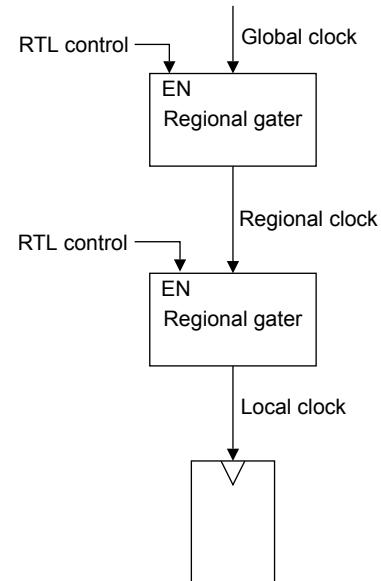


Figure 2-31 Clocking hierarchy

2.2.4 Clock enable inputs

CMN-700 includes several clock enable inputs.

Note

The following description describes the relationship of the clock enable inputs to **GCLK0**. If your configuration uses multiple asynchronous clock domains, then the same relationship applies to the individual global clock signals for each clock domain.

The clock enable input signals are:

- | | |
|-----------------|--|
| ACLKEN_S | This input is present on each AMBA slave interface. |
| ACLKEN_M | This input is present on each AMBA master interface. |
| ATCLKEN | This input is present on each debug and trace ATB interface. |

All clock enables, shown here as ***CLKEN***, have identical functionality, enabling the respective interfaces with which they are included to run at integer fractions of **GCLK0**. In other words, the clock enables run slower than **GCLK0**, ranging from ratios of 1:1 to 4:1. **ATCLKEN** is limited to 1:1, 2:1, and 4:1 integer fractions. This approach enables synchronous communication with slower SoC logic.

CLKEN asserts one **GCLK0** cycle before the rising edge of **SoC-CLK**. SoC control logic can change the ratio of **GCLK0** frequency to the SoC clock, **SoC-CLK**, frequency dynamically using ***CLKEN***.

The following figure shows a timing example of a ***CLKEN*** ratio change. In the example, ***CLKEN*** changes the ratio of the relevant interface frequency respective to **GCLK0** from 3:1 to 1:1.

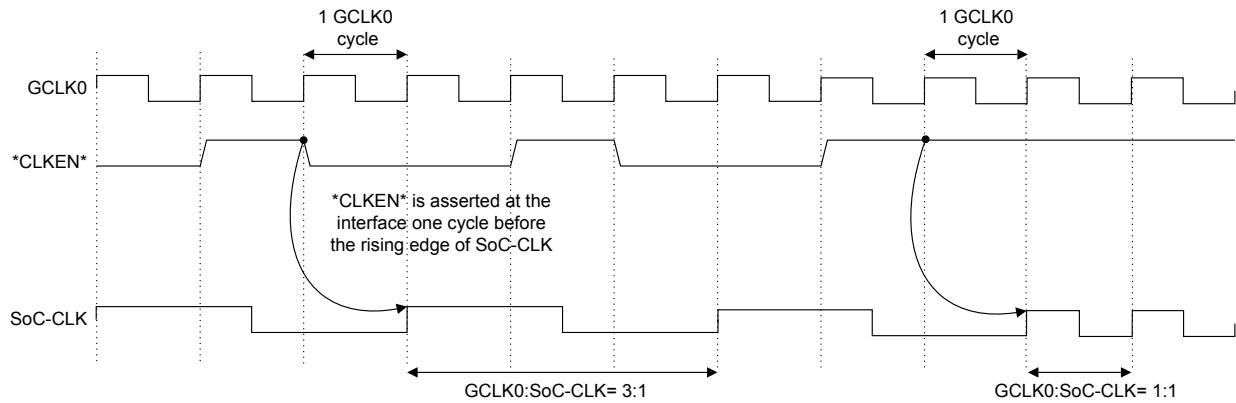


Figure 2-32 *CLKEN* with GCLK0:SoC-CLK ratio changing from 3:1 to 1:1

2.2.5 High-level Clock Gating

The PCCB supports a *High-level Clock Gating* (HCG) mechanism. This mechanism notifies the SoC when CMN-700 is inactive and therefore reduces dynamic power consumption.

HCG enables an external SoC clock control unit, the *External Clock Controller* (ExtCC), to stop the **GCLKn** clock inputs. For more information about the ExtCC, see [2.2.6 External Clock Controller](#) on page [2-79](#).

————— Note ————

If your CMN-700 configuration has multiple asynchronous clock domains, the ExtCC stops each individual global clock signal identically.

CMN-700 includes a Q-Channel interface that enables CMN-700 and the SoC to communicate to achieve HCG functionality through the PCCB. For more information, see the *AMBA® Low Power Interface Specification Arm® Q-Channel and P-Channel Interfaces*.

2.2.6 External Clock Controller

The ExtCC is used to control the clock gating flow.

The following figure shows an example of how the ExtCC controls the clock gating flow. This example clock gating sequence begins and ends with the Q-Channel in either of the following states:

- Quiescent state (**Q_STOPPED**), where **QREQn** and **QACCEPTn** are asserted
- Active state (**Q_RUN**), where **QREQn** and **QACCEPTn** are deasserted

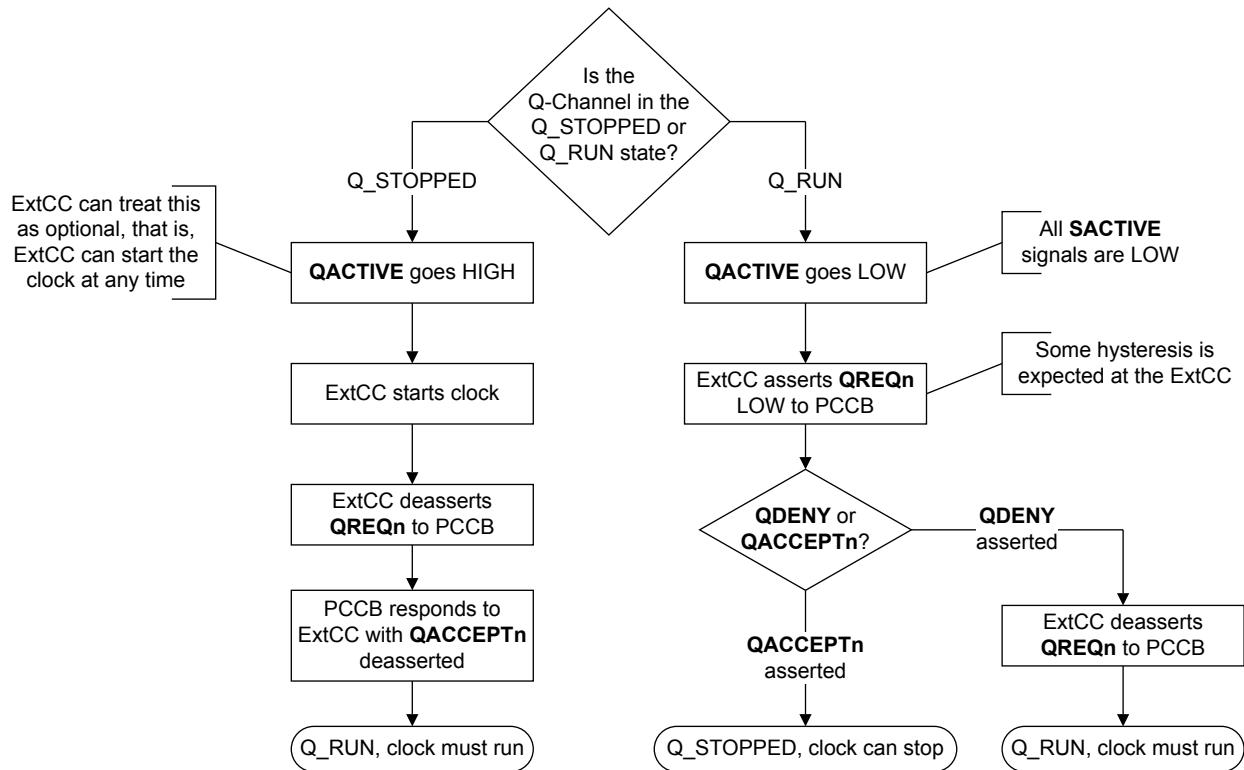


Figure 2-33 Clock gating control using ExtCC

The requirements of the ExtCC are as follows:

- It must supply a clock to CMN-700 when the Q-Channel is in any state other than Q_STOPPED.
- The ExtCC can either choose to gate the clock to CMN-700 when the Q-Channel is in the Q_STOPPED state, or it can choose to run the clock at any time.
- ExtCC is responsible for bringing the Q-Channel to Q_RUN state after reset deassertion.
- Although this manual does not describe the exact behavior of the ExtCC and its usage of **QREQn** in response to **QACTIVE** deassertion, the design of the ExtCC is likely to include a control loop with some hysteresis. Therefore HCG is enabled when the system is inactive for long periods, but is not enabled for short periods of inactivity. If the clocks are stopped in response to short periods of inactivity, performance of CMN-700 can be negatively affected.
- It is the responsibility of the SoC designer to fully control the clock management Q-Channel. If there is a requirement for a control or configuration bit to completely enable or disable HCG functionality, that register or bit must exist outside of CMN-700. More specifically, CMN-700 has no internal means of disabling HCG.

2.2.7 CXG clock management

CMN-700 CML/CXG configurations add one or two extra clock domains, depending on whether you configure the CXS logic to be synchronous or asynchronous to **GCLKn**.

Note

CLK_CGL can drive **CLK_CXS** in synchronous configurations. For more information, see [2.2.2 CXG clock inputs on page 2-75](#).

Synchronous or asynchronous configuration is based on the **CXLADB_PRESENT** parameter.

CML/CXG configurations add Q-Channel interfaces for each CXG instance and corresponding CXS interface:

CLK_CGL Q-Channel

Manages the CGL link and CGL domain logic in the CXG and CXLA devices.

CLK_CXS Q-Channel

Manages the CXS link interface and CXS clock domain logic in the CXLA.

The following table shows the possible clock states for the CMN-700 and CML device clocks, where N denotes multiple CXS interfaces:

Table 2-9 CMN-700 and CML device clock states

GCLKn	CLK_CGL[N]	CLK_CXS[N]	Description
RUN	RUN	RUN	CMN-700 and CXS[N] interface active
RUN	RUN	STOP	CXS[N] domain gated. This state is only valid for asynchronous configuration where CLK_CGL is asynchronous to CLK_CXS .
RUN	STOP	RUN	CGL[N] inactive, transitory state. This state is only valid for asynchronous configuration where CLK_CGL is asynchronous to CLK_CXS .
RUN	STOP	STOP	CXS[N] interface fully gated
STOP	STOP	RUN	CXS[N] active, others inactive, transitory state. This state is only valid for asynchronous configuration where CLK_CGL is asynchronous to CLK_CXS .
STOP	STOP	STOP	CMN-700 fully gated, all CXS[N] interfaces inactive

————— **Note** —————

If your configuration has multiple asynchronous clock domains, then the information in the preceding table applies equally to each global clock signal.

CML CCG block runs at CMN-700 clock domain (GCLKn). It adds Q-Channel interface for each CCG instance:

CLK_CXS Q-Channel

Manages the CXS link interface.

————— **Note** —————

Optionally, CXS Domain Bridge (CXSDB) can be added on each of these CXS interfaces (IssueB) to operate CXS interface asynchronously to GCLKn. This CXSDB sits outside CCG block.

2.2.8 Reset

CMN-700 has a single global reset input signal, **nSRESET**.

nSRESET is an active-LOW signal that can be asynchronously or synchronously asserted and deasserted.

When asserted, **nSRESET** must remain asserted for 90 clock cycles. Likewise, when deasserted, **nSRESET** must remain deasserted for 90 clock cycles. This requirement ensures that all internal CMN-700 components enter and exit their reset states correctly.

All CMN-700 clock inputs must be active during the required 90-cycle, or larger, period of **nSRESET** assertion. The clock inputs must also remain active for at least 90 cycles following deassertion of **nSRESET**.

2.2.9 CXG reset

There are two extra reset inputs for the CML/CXG configuration: **nRESET_CGL** and **nRESET_CXS**.

Both **nRESET_CGL** and **nRESET_CXS** are active-LOW signals that can be asynchronously or synchronously asserted and deasserted.

When asserted, **nRESET_CGL** and **nRESET_CXS** must remain asserted for 20 **CLK_CGL** and **CLK_CXS** clock cycles respectively. Likewise, when deasserted, **nRESET_CGL** and **nRESET_CXS** must remain deasserted for 20 **CLK_CGL** and **CLK_CXS** clock cycles respectively. This requirement ensures that all CML components enter and exit their reset states correctly.

Both **CLK_CGL** and **CLK_CXS** must be active during the required 20-cycle, or larger, period of **nRESET_CGL** and **nRESET_CXS** assertion respectively and must remain active for at least 20 cycles following deassertions.

For more relationship information between the CXS and CGL domains, refer to [2.2.2 CXG clock inputs on page 2-75](#).

————— Note ————

There is no sequencing requirement between the CMN-700 **nSRESET** and the CML resets. However, the CML domains must exit reset before CXLA functionality is required.

2.3 Power management

CMN-700 includes several power management capabilities, that are either externally controllable or are assisted by the SoC.

CMN-700 has the following power management capabilities:

- Several distinct predefined power states. These states include ones in which all, half, or none of the SLC Tag and Data RAMs can be powered up, powered down, or in retention:
 - A state in which only the HN-F SF is active
 - A state in which the SLC RAMs and SF RAMs are inactive

These power states reduce static and dynamic power consumption.

- Support for static retention in HN-F in which the SoC places SLC and SF RAMs in a retention state. This capability reduces static power consumption.
- Support for in-pipeline low-latency Data RAM retention control, in which a programmable idle counter can be used to put the SLC RAMs in retention.

————— Note —————

The clocking hierarchy and clock gating mechanism are described in [2.2 Clocks and resets on page 2-73](#).

This section contains the following subsections:

- [2.3.1 Power domains on page 2-83](#).
- [2.3.2 Power domain control on page 2-85](#).
- [2.3.3 P-Channel on device reset on page 2-86](#).
- [2.3.4 CXG power domain on page 2-86](#).
- [2.3.5 HN-F Memory retention mode on page 2-87](#).
- [2.3.6 HN-F power domains on page 2-87](#).
- [2.3.7 HN-F RAM PCSM Interface on page 2-91](#).
- [2.3.8 HN-F power domain completion interrupt on page 2-91](#).
- [2.3.9 RN entry to and exit from Snoop and DVM domains on page 2-92](#).

2.3.1 Power domains

The power domains in CMN-700 are split between the logic and RAMs within the HN-F partitions.

The power domains are:

Logic

All logic except HN-F SLC Tag and Data RAMs and HN-F SF RAMs.

System Level Cache RAM0

SLC Tag and Data RAMs way[7:0] within HN-F partitions. The RAMs in each HN-F partition can be independently controlled.

System Level Cache RAM1

SLC Tag and Data RAMs way[15:8] within HN-F partitions. The RAMs in each HN-F partition can be independently controlled. The RAM1 domain for 3MB SLC size configurations includes way[11:8].

Snoop filter only mode

SF RAMs within HN-F partitions. The RAMs in each HN-F partition can be independently controlled.

The following figure shows an example power domain configuration.

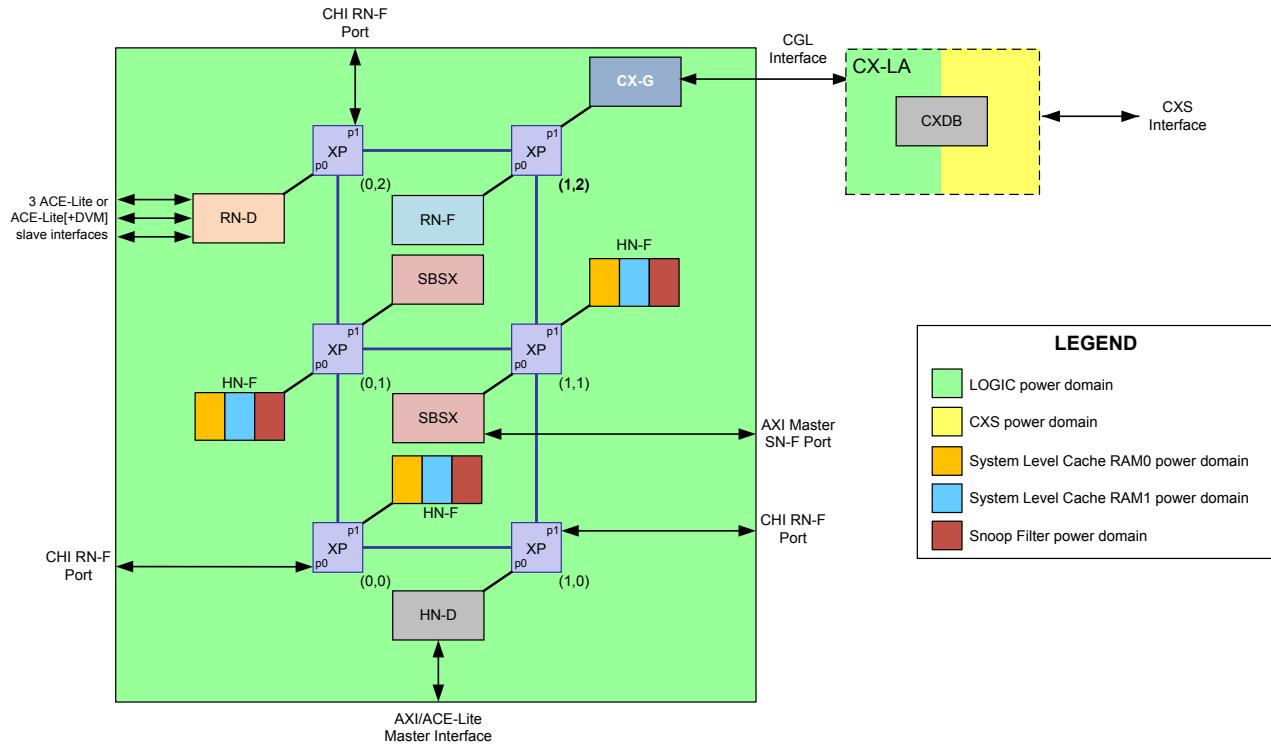


Figure 2-34 CMN-700 power domain example

The following figure shows another example power domain configuration, where the CXLA component is in the same power domain.

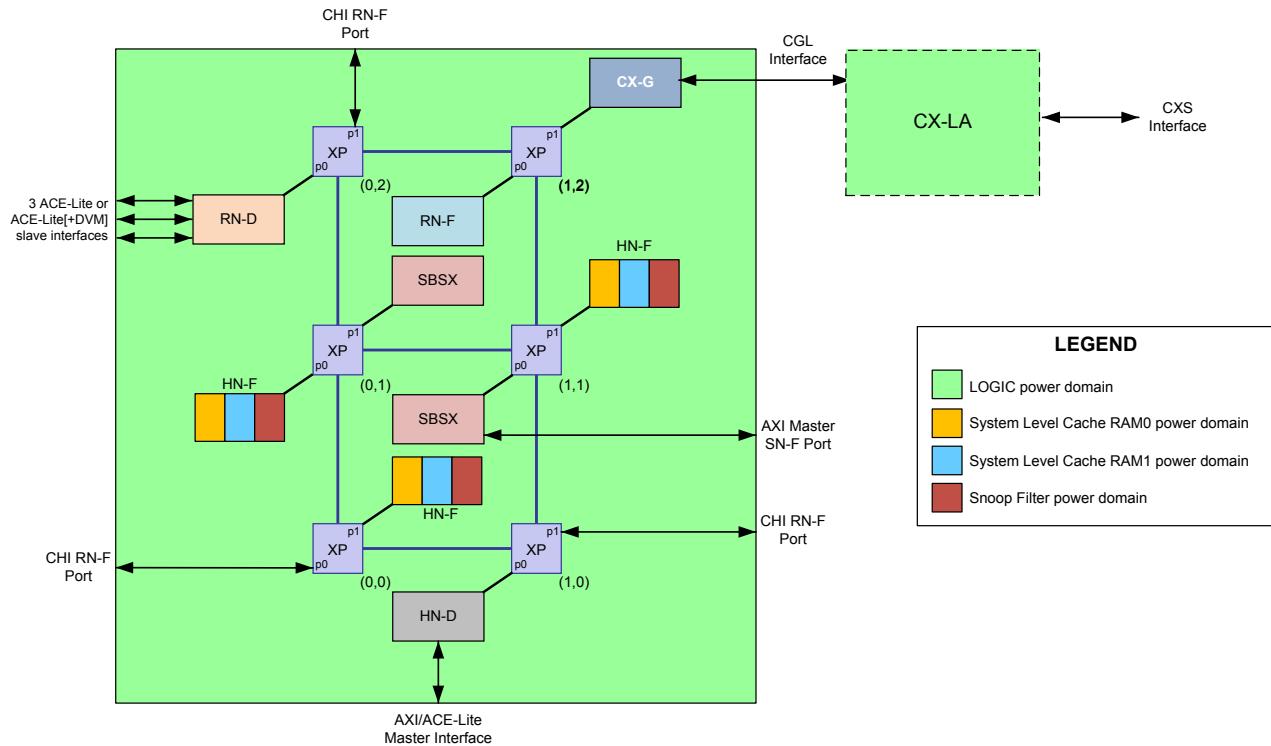


Figure 2-35 Single CML power domain example

————— Note ————

MTU RAMs within MTSX are in the logic domain and cannot be independently controlled.

2.3.2 Power domain control

The CMN-700 Logic P-Channel controls all power domains except for the RAM and CXS power domains.

In addition to controlling the Logic domain, the Logic P-Channel allows synchronization between the HN-F software-controlled power domains and the Logic domain. This synchronization is achieved through a CONFIG state, as the following figure shows.

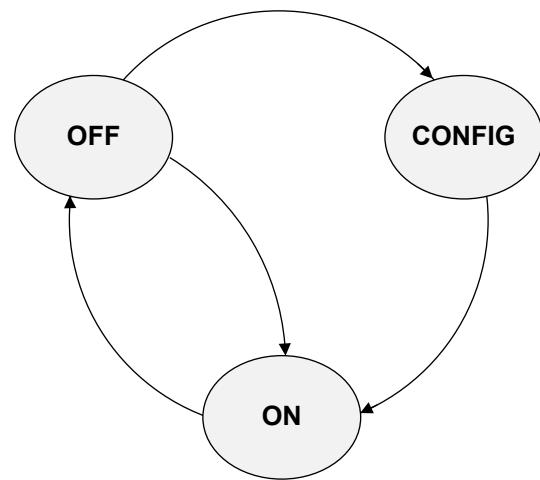


Figure 2-36 Logic domain states

There are two paths for transitioning from the OFF to ON state:

Cold reset

The Logic PSTATE OFF to ON transition also initiates NOSFSLC to FAM transition for all HN-F partitions.

Exit from HN-F Static Retention state

The Logic PSTATE transitions from OFF to CONFIG, indicating that CMN-700 is exiting a Memory Retention state, and does not initiate any HN-F partition power transitions.

The following table contains the power modes of components within the domain and the associated PSTATE values.

Table 2-10 Power mode configurations and PSTATE values

Power mode	PSTATE	CMN-700 Logic	HN-F
OFF	0b00000	OFF	OFF/MEM_RET
CONFIG	0b11000	ON	ANY
ON	0b01000	ON	ANY

For an introduction to HN-F states, see [2.3.6 HN-F power domains on page 2-87](#).

For P-Channel signal list information, see [B.10 Power management signals on page Appx-B-1395](#).

CXS power gating

CML/CXG systems contain an extra power domain, the CXS power domain, which is present according to the value of the CXLA DB_PRESENT parameter. CXS logic is controlled by a combination of the CXS Power Q-Channel and the CXS Q-Channel that controls the clocks. Unless both the CXS Power Q-Channel and the CXS Q-Channel are in the OFF State, power must be provided.

2.3.3 P-Channel on device reset

This section shows how to initialize the power state of a power domain.

Certain device power states might power down the control logic. When powering this control logic back on, the power controller must indicate the state that the device must power up. The device detects the required state by sampling **PSTATE** when **nSRESET** deasserts. The **PSTATE** inputs must be asserted before the deassertion of reset and remain after the deassertion of **nSRESET**, to allow reset propagation within CMN-700. The power controller must ensure that the reset sequence is complete before transitioning **PSTATE**, otherwise the device might sample an undetermined value. The following figure shows the state initialization on reset while **PREQ** is deasserted. CMN-700 also supports P-Channel initialization with **PREQ** asserted at **nSRESET** deassertion.

————— Note —————

PSTATE inputs must be static 100 cycles before deassertion of **nSRESET**, and also for 100 cycles after the deassertion of **nSRESET**.

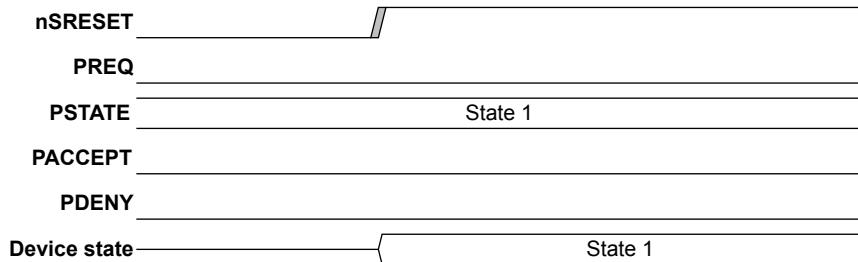


Figure 2-37 Reset state initialization with PREQ deasserted

For an introduction to HN-F states, see [2.3.6 HN-F power domains on page 2-87](#).

2.3.4 CXG power domain

CXG/CXS interfaces contain an extra power domain, the CXS power domain. The CMN-700 CXS Power Q-Channel controls the CXS power domain.

The CXS power domain is present depending on the value of the CXLA DB_PRESENT parameter.

————— Note —————

If the CXS interface is inactive, the CXS power domain can be shut off.

The following table shows the possible CMN-700 LOGIC and CXS power states, where N denotes multiple CXS interfaces:

Table 2-11 CMN-700 LOGIC and CXS power states

LOGIC state	CXS[N] state	Description
ON	ON	CMN-700 and CXS[N] interface active.
ON	OFF	CXS[N] interface inactive.

Table 2-11 CMN-700 LOGIC and CXS power states (continued)

LOGIC state	CXS[N] state	Description
OFF	ON	CXS[N] domain active, transitory state.
OFF	OFF	Shut down, all CXS[N] interfaces inactive.

2.3.5 HN-F Memory retention mode

When isolating the CMN-700 outputs, handshake protocols on certain interfaces must be followed. This section describes the steps to take to enter and exit HN-F Memory retention mode.

Entering HN-F Memory retention mode

1. Program the HN-Fs to enter the required power state.
2. Quiesce the interconnect, and wait for QACTIVE to drop.
3. Place CMN-700 in LOGIC_OFF state through the Logic P-Channel.
4. Isolate the CMN-700 outputs. If the logic on the other side of the interface is being powered down or reset, this step might not be needed.
5. Turn off power to CMN-700.

Exiting HN-F Memory retention mode

1. Apply power to CMN-700.
2. Assert reset.
3. Enable clocks.
4. Disable isolation of the CMN-700 outputs.
5. Deassert reset.
6. Place CMN-700 in LOGIC_CONFIG state through the logic P-Channel.
7. Reprogram the HN-F PWPR to the retention mode the HN-F was in before turning off power.
8. Reprogram the HN-F PWPR to ON.
9. Reprogram the CMN-700 configuration registers, including the RN SAM and any other registers written during cold boot.
10. Place CMN-700 in LOGIC_ON state through the P-Channel.
11. Resume traffic/normal operation.

2.3.6 HN-F power domains

The HN-F has various power states. Transitioning between different states enables or disables different parts of the HN-F.

The HN-F has three classes of power states:

1. Operational states, where logic is on and enabled RAMs are operating as normal.
2. Functional Retention states, where logic is on, and enabled RAMs are in retention.
3. Memory Retention states, where logic is off and enabled RAMs are in retention.

Within these power states, the HN-Fs in an SCG operate in four modes:

FAM *Full Associativity Mode* (FAM), where the SF and the entire SLC are enabled.
HAM *Half-Associativity Mode* (HAM), where the SF is enabled but the upper half of the SLC ways are disabled and powered off.

SFONLY *Snoop filter only mode* (SFONLY), where the SF is enabled but the whole SLC is powered off.

NOSFSLC *No-SLC Mode* (NOSFSLC), where the SF and SLC are disabled and powered off.

The following constraints apply to the power states and transitions:

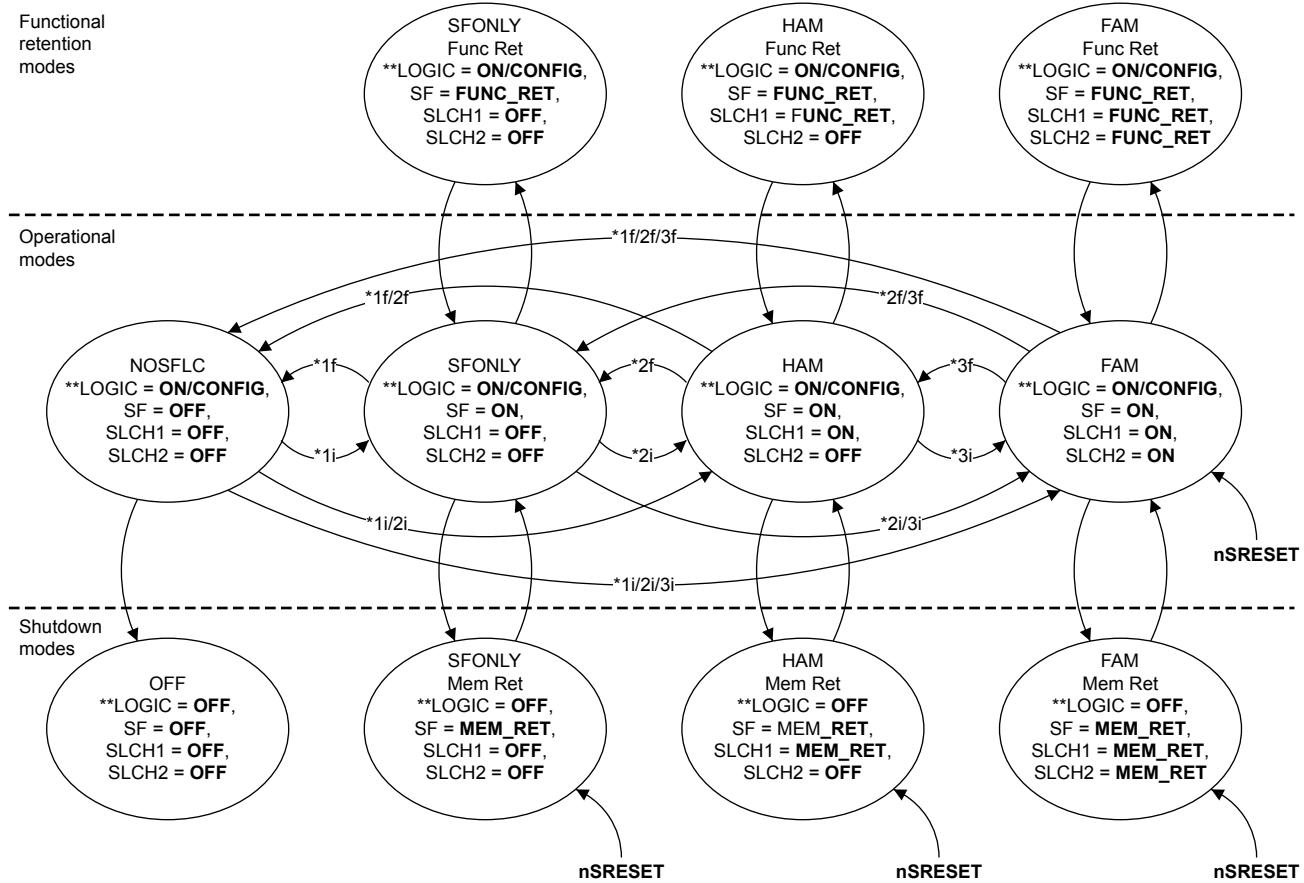
- If SLC size is 0KB, the HN-F does not support transitions to FAM or HAM modes.
- After initialization, the power status register indicates FAM instead of SFONLY for 0KB SLC configurations.
- When a power transition is initiated, another must not be initiated until the first one completes.

The following table shows the valid HN-F power states and their requirements.

Table 2-12 HN-F power states

State	Description	Control logic	SF power state	SLC way[7:0] power state	SLC way[15:8] power state
FAM	Full Run mode	On	On	On	On
HAM	Run mode with SLCH2 (SLC upper ways) disabled	On	On	On	Off
SF	Run mode with SLCH1 and SLCH2 disabled	On	On	Off	Off
NOSFSLC	Run mode with SLCH1, SLCH2, and SF disabled	On	Off	Off	Off
FAM FUNC_RET	Run mode with SLCH1, SLCH2, and SF in dynamic retention	On	Retention	Retention	Retention
HAM FUNC_RET	Run mode with SLCH1 and SF in retention, and SLCH2 in power down	On	Retention	Retention	Off
SF FUNC_RET	Run mode with SF in retention, and SLCH1 and SLCH2 in power down	On	Retention	Off	Off
FAM MEM_RET	Shut down with SLCH1, SLCH2, and SF in retention	Off	Retention	Retention	Retention
HAM MEM_RET	Shut down with SLCH1 and SF in retention, and SLCH2 in power down	Off	Retention	Retention	Off
SF MEM_RET	Shut down with SF in retention, and SLCH1 and SLCH2 in power down	Off	Retention	Off	Off
OFF	Shutdown	Off	Off	Off	Off

The following figure shows the valid power states and transitions for a CMN-700 system.



Note: **BOLD** text shows the required power state.

* Automatic initialization and flushing actions:

- 1i: Initialize snoop filter RAMs.
- 2i: Initialize lower ways of tag RAMs.
- 3i: Initialize upper ways of tag RAMs.
- 1f: Flush (force back-invalidations as necessary and invalidate) snoop filter RAMs.
- 2f: Flush (clean/invalidate) lower ways of tag/data RAMs.
- 3f: Flush (clean/invalidate) upper ways of tag/data RAMs.

** All designations refer to P-state values required to enter the respective state.

Figure 2-38 Power state transitions

The SF does not track RN-F coherence while the HN-F is in NOSFSLC state. Therefore, the RN-F caches must be flushed before transitioning from NOSFSLC to SFONLY, HAM, or FAM states.

These HN-F power states are transitioned using configuration register writes that must target all HN-Fs in the SCG region. Also, the logic domain P-Channel interface can initiate a NOSFSLC→FAM transition.

Write to the following por_hnf_ppu_pwpr register fields to transition HN-F partitions to a required power state:

- policy
- op_mode

When the power state transition is complete, the following por_hnf_ppu_pwsr register fields are updated:

- pow_status
- op_mode_status

If either the SLC, the SF, or both are flushed as part of a power transition, then the power state transition can take many thousands of clock cycles. Also, the INTREQPPU interrupt output can be used to indicate the completion of the HN-F power state transitions.

From the FAM, HAM, or SFONLY modes, the HN-F can enter a dynamic retention mode using configuration register writes, where:

- The logic power is on.
- The voltage to the RAMs is on, but is reduced to a level that is sufficient for bitcell retention but insufficient for normal operation.
- The array pipeline is blocked, and a handshake occurs to allow array access when exiting the retention state.

These dynamic power transitions are executed autonomously within each HN-F partition. Each HN-F has a programmable idle cycle counter and initiates a P-Channel handshake with the corresponding RAMs to enter the dynamic retention state. The pipeline then blocks transactions that target the HN-F RAMs. A coherent transaction triggers an exit from the dynamic retention state, initiates another P-Channel handshake, and takes the RAMs out of dynamic retention mode.

From these states, the SLC can also enter a Memory retention mode, where:

- The logic power is turned off.
- The voltage to the RAMs is on, but is reduced to a level that is sufficient for bitcell retention but insufficient for normal operation.
- Reset deassertion is essential when exiting retention after logic power down.

A P-Channel interface controls the CMN-700 logic domain power state. This P-Channel interface also interacts with the HN-F power control logic using an internal bus. The HN-F power control logic waits for a command on the deassertion of **nSRESET**, depending on the overall power state transition that is required. For the Cold reset HN-F FAM transition case, the PCCB block initiates the HN-F NOSFSLC→FAM command. To exit static retention cases, the SCP initiates configuration register writes to the HN-F to indicate the HN-F power state.

The circumstances where the HN-F enters dynamic retention modes or static retention modes are different. Dynamic retention is entered because of a dynamic activity or inactivity indicator from the HN-F to the SoC. This indicator is an output of the HN-F, and is used to determine periods of inactivity long enough to warrant entering retention mode. However the inactivity is either not long enough or not the type of inactivity to make the SoC place the SLC and SF into static retention. In addition to the static retention modes, the control logic can be powered down from the NOSFSLC state, at which point CMN-700 is fully off.

The HN-Fs perform all activity that is required to enable safe transition between the respective power states automatically in response to input P-Channel PSTATE transitions. The SoC logic does not need to perform any additional activity to enable transitions between power states. For example, the HN-F performs clean and invalidation of half of the ways of the SLC and clean and invalidation of all ways of the SLC. This clean and invalidation activity occurs as required by the respective power state transitions.

Note

CMN-700 cannot make any power transitions while the control logic is powered off. Consider a transition from FAM static retention to OFF. To complete this transition, the power state must first move through FAM and NOSFSLC states while the LOGIC power domain is on. These transitions allow the SLC and SF to be flushed.

The following table shows the PSTATE encodings for the HN-F and power domains including RAM configurations for the different operational modes.

————— Note ————

HN-F cannot process any transactions while in static retention (FUNC_RET or MEM_RET). HN-F must be in the ON state before sending any transactions to HN-F in this case. If HN-F is in dynamic retention, any activity autonomously takes HN-F out of dynamic retention.

Table 2-13 Power modes, operational modes, and RAM configurations

Operational mode	Power mode	PSTATE	Bank 0 RAM	Bank 1 RAM	SF RAM
FAM	ON	11_1000	ON	ON	ON
	FUNC_RET	11_0111	RET	RET	RET
	MEM_RET	11_0010	RET	RET	RET
HAM	ON	10_1000	ON	OFF	ON
	FUNC_RET	10_0111	RET	OFF	RET
	MEM_RET	10_0010	RET	OFF	RET
SFONLY	ON	01_1000	OFF	OFF	ON
	FUNC_RET	01_0111	OFF	OFF	RET
	MEM_RET	01_0010	OFF	OFF	RET
NOSFSLC	MEM_OFF	00_0110	OFF	OFF	OFF
	OFF	00_0000	OFF	OFF	OFF

2.3.7 HN-F RAM PCSM Interface

Each HN-F RAM interface contains a *Power Control State Machine* (PCSM).

Each PCSM P-Channel interface that can be used to convert power state transitions into technology-specific controls, and the overall HN-F partition power state transition, depends on all P-Channel transactions to complete.

The following table lists the valid PSTATE values for this interface.

Table 2-14 PSTATE Encodings

PSTATE	Value
ON	0b1000
FUNC_RET	0b0111
MEM_RET	0b0010
OFF	0b0000

————— Note ————

This interface does not have a **PDENY** signal.

2.3.8 HN-F power domain completion interrupt

The PCCB can be configured to generate interrupt INTREQPPU on completion of power state transitions for a collection of HN-Fs.

The PCCB contains a global status register, por_ppu_int_status, which indicates HN-F power state transition completion. The PCCB also contains a mask register, por_ppu_int_mask, which allows filtering on all or a subset of the HN-Fs in the CMN-700 configuration. The bit positions in the por_ppu_int_status and por_ppu_int_mask registers correspond to the logical ID of the HN-Fs.

INTREQPPU asserts when all por_ppu_int_status register bits with the corresponding por_ppu_int_mask register bit are set by the HN-F power transition completion.

To deassert **INTREQPPU**, write 0b1 to the bits of the por_ppu_int_status register that correspond to the masked group of HN-Fs that completed the power transitions.

2.3.9 RN entry to and exit from Snoop and DVM domains

CMN-700 includes a feature that allows RNs to be included or excluded from the system coherency domain. This domain is also known as the Snoop domain or DVM domain. This feature ensures correct operations of Snoops and DVMs when:

- An RN is taken out of reset.
- An RN is powered down and then later powered up.

RN-Fs behave as follows:

- If an RN-F is included in the system coherency domain, it must respond to Snoop and DVM requests from CMN-700.
- If an RN-F is excluded from the system coherency domain, it does not receive Snoop or DVM requests from CMN-700.

RN-Ds behave as follows:

- If an RN-D is included in the system coherency domain, it must respond to DVM requests from CMN-700.
- If an RN-D is excluded from the system coherency domain, it does not receive DVM requests from CMN-700.

Hardware interface

This section describes the hardware interface for RN inclusion into and exclusion from system coherency domain.

CMN-700 provides two signals for RN system coherency entry and exit:

- **SYSCOREQ** input (to CMN-700)
- **SYSCOACK** output (from CMN-700)

These two signals implement a four-phase handshake between the RN and CMN-700 with the four states as specified in the following table.

Table 2-15 RN system coherency states

SYSCOREQ	SYSCOACK	State
0	0	DISABLED
1	0	CONNECT
1	1	ENABLED
0	1	DISCONNECT

Coming out of Reset, the RN is in the DISABLED system coherency state.

CONNECT

To enter system coherency, RN must assert **SYSCOREQ** and transition to CONNECT state. The RN must be ready to receive and respond to Snoop and DVM requests in this state.

ENABLED

Next, CMN-700 asserts **SYSCOACK** and transitions to ENABLED state. The RN is now included in the system coherency domain. The RN can receive and must respond to Snoop and DVM requests in this state.

DISCONNECT

When the RN is ready to exit system coherency, it must deassert **SYSCOREQ** and transition to DISCONNECT state. The RN continues to receive and must respond to Snoop and DVM requests in this state.

DISABLED

When all outstanding Snoop and DVM responses have been received, CMN-700 deasserts **SYSCOACK** and transitions to DISABLED state. In this state, no snoop or DVM transactions are sent to the RN which can now be powered down.

Note

The following rules must be obeyed to adhere to the four-phase handshake protocol.

- When **SYSCOREQ** is asserted, it must remain asserted until **SYSCOACK** is asserted.
 - When **SYSCOREQ** is deasserted, it must remain deasserted until **SYSCOACK** is deasserted.
-

Software interface

This section describes the software interface for RN inclusion into and exclusion from system coherency domain.

CMN-700 provides two *Configuration Registers* (CRs) for system coherency entry and exit:

- | | |
|-------------|--|
| RN-F | <ul style="list-style-type: none">• por_mxp_p_0-5_syscoreq_ctl on page 3-1012• por_mxp_p_0-5_syscoack_status on page 3-1013 |
| RN-D | <ul style="list-style-type: none">• por_rnd_syscoreq_ctl on page 3-273• por_rnd_syscoack_status on page 3-274 |

Reading and writing to these CRs provides a software alternative to the 4-phase hardware handshake.

Note

It is possible the CRs contain multiple bits where each bit corresponds to a different RN. The following description is about the Read and Write of the CR bit that corresponds to a given RN. When configuring the system coherency entry or exit for a given RN, software must adopt a Read-Modify-Write strategy. This strategy ensures CR bits corresponding to other RNs, are not modified when writing into the syscoreq_ctl CR.

Coming out of Reset, both CRs are cleared, indicating DISABLED state.

CONNECT

To initiate an RN entry into the system coherency domain, software must first poll both CRs. This poll ensures the CR bits corresponding to that RN, are set to **0b0**. When the RN is ready to receive and respond to Snoop and DVM requests, software must write a **0b1** into the corresponding bit in the syscoreq_ctl CR. This write process transitions the RN to CONNECT state.

ENABLED

Next, CMN-700 indicates a transition to ENABLED state by setting the corresponding CR bit in the syscoack_status register to **0b1**. The RN is now inside the system coherency domain. Software can poll the syscoack_status register to determine this state transition.

DISCONNECT

To initiate an RN exit from the system coherency domain, software must first poll both CRs.

After ensuring that the CR bits corresponding to that RN are set, software must clear the corresponding syscoreq_ctl bit to transition the RN to DISCONNECT state. The RN continues to receive and must respond to Snoop and DVM requests in this state.

DISABLED

When all outstanding Snoop and DVM responses have been received, CMN-700 clears the corresponding syscoack_status bit indicating the transition to DISABLED state. In this state, no Snoop or DVM transactions are sent to the RN. Software must poll the syscoack_status register to ensure that this state transition has occurred before initiating RN powerdown.

————— Note ————

To adhere to the four-phase handshake protocol, the following rules apply:

- The hardware interface is intended as the primary interface. The software interface is provided as an alternative for legacy RN devices and systems that do not support the hardware interface.
- Either hardware or software interface must be used, but not both. Coming out of Reset, the hardware interface is enabled by default. The first write into the syscoreq_ctl register disables hardware interface and enables software interface. There must be a Reset to re-enable hardware interface.
- When software interface is employed, **SYSCOREQ** must remain deasserted.
- When hardware interface is employed, software must not write to the syscoreq_ctl CR.

2.4 Network layer functions

CMN-700 has specific functions that it uses to map regions of the address space, determine flit targets, and define overall routing behavior. Some of these functions are configurable by setting configuration parameters or by software.

This section contains the following subsections:

- [2.4.1 Node ID mapping on page 2-95](#).
- [2.4.2 Node ID mapping for configurations with extra device ports on page 2-99](#).
- [2.4.3 Addressing capabilities on page 2-103](#).
- [2.4.4 System Address Map on page 2-104](#).
- [2.4.5 RN SAM on page 2-105](#).
- [2.4.6 CXRA/CCRA SAM on page 2-124](#).
- [2.4.7 HN-F SAM on page 2-125](#).
- [2.4.8 SAM memory region size configuration on page 2-136](#).
- [2.4.9 HN-I SAM on page 2-138](#).
- [2.4.10 GIC communication over AXI4-Stream ports on page 2-146](#).
- [2.4.11 Default XY routing behavior on page 2-147](#).
- [2.4.12 Non-XY routing on page 2-148](#).
- [2.4.13 Extended CCIX Requesting Agent ID mechanism for up to 512 RN-Fs on page 2-153](#).
- [2.4.14 Cross chip routing and ID mapping on page 2-155](#).
- [2.4.15 CCIX1.1 port-to-port forwarding on page 2-161](#).

2.4.1 Node ID mapping

The physical position of a device in the mesh determines the node ID mapping.

This scheme is the default node ID-mapping scheme. This scheme is different from the one that is used when using extra device ports on MXPs. For more information about that scheme, see [2.4.2 Node ID mapping for configurations with extra device ports on page 2-99](#).

The following details determine the physical position of a device in the mesh:

1. The X coordinate of its XP
2. The Y coordinate of its XP
3. The XP device port (0 or 1) that it connects to

The device node ID is mapped to (X, Y, Port, `0b00`).

————— Note —————

1. The bit widths of the X and Y parameters depend on the configured size of the mesh.
2. The naming convention for I/O signals uses decimal values of the node ID. For example, `RXREQFLIT_NIDxxx` uses `xxx` values in decimal.

The node ID size depends on the X and Y dimensions of the CMN-700 mesh. The larger of the X and Y dimensions determines the size, as the following table shows.

Table 2-16 Node ID size selection

X mesh dimension	Y mesh dimension	Node ID size
4 or less	4 or less	7 bits
5-8	8 or less	9 bits
8 or less	5-8	
9 or more	9 or more	11 bits

Note

On internal CHI interfaces, the NodeID width is set equal to the NodeID size. On external CHI interfaces, the NodeID width is set to 11 bits, where unused bits occupy MSBs and are driven to zeroes.

The following tables contain the different node ID formats.

Table 2-17 7-bit node ID format

[6:5]	[4:3]	[2]	[1:0]
X position	Y position	Port	0b00

Table 2-18 9-bit node ID format

[8:6]	[5:3]	[2]	[1:0]
X position	Y position	Port	0b00

Table 2-19 11-bit node ID format

[10:7]	[6:3]	[2]	[1:0]
X position	Y position	Port	0b00

The following figure shows a CMN-700 system with 7-bit node IDs in (X, Y, Port, DeviceID) format.

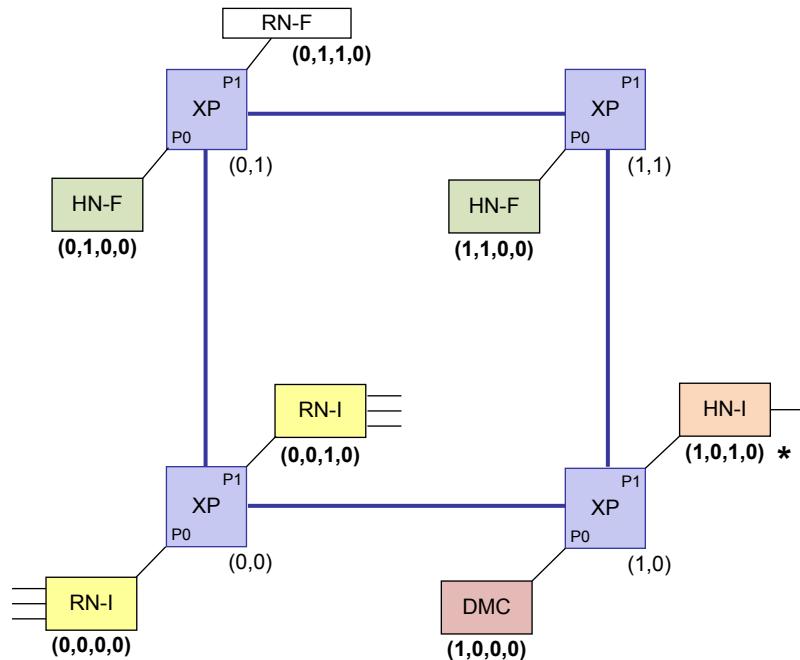


Figure 2-39 Example system with 7-bit node IDs

Example 2-2 7-bit node ID format

For the HN-I connected to XP (1,0), the node ID reads as (1,0,1,0).

This format is equivalent to (0b01, 0b00, 0b1, 0b00) or 0x24.

If CAL is present, the two devices that are connected to the CAL are assigned consecutive node IDs. One device is assigned NodeID[1:0]=0b00 and the other device is assigned NodeID[1:0]=0b01.

The following figure shows a CMN-700 system with CAL and 7-bit node IDs in (X, Y, Port, DeviceID) format.

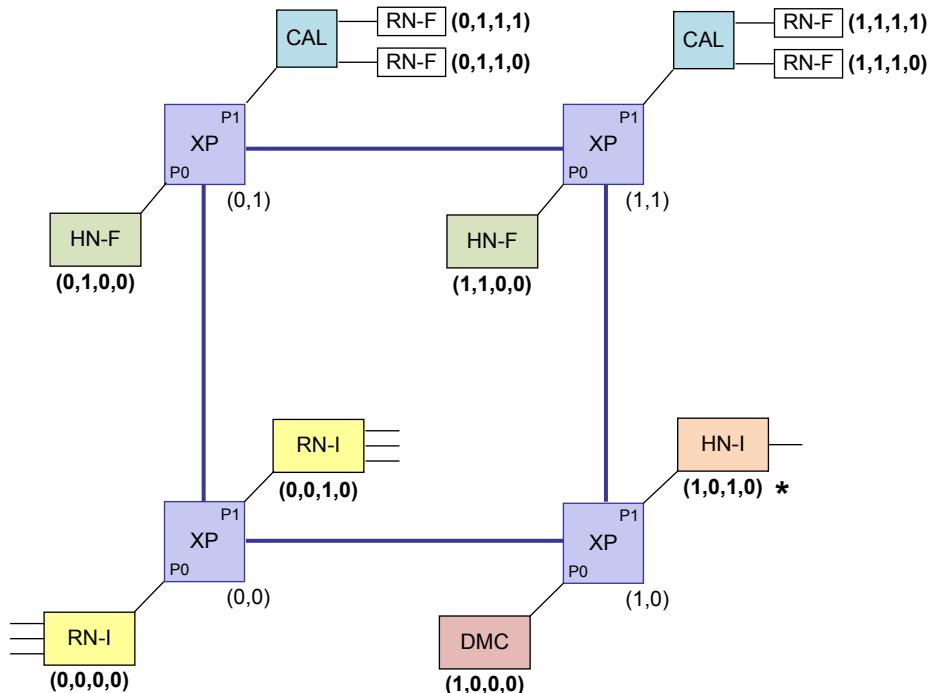


Figure 2-40 Example system with 7-bit node IDs and CAL

If CAL is present, the devices that are connected to the CAL are assigned consecutive node IDs. For a CAL2, one device is assigned NodeID[1:0]=0b00 and the other device is assigned NodeID[1:0]=0b01. For a CAL4:

- Device 0 NodeID[1:0] = 0b00
- Device 1 NodeID[1:0] = 0b01
- Device 2 NodeID[1:0] = 0b10
- Device 3 NodeID[1:0] = 0b11

The following figure shows a CMN-700 system with CAL2 and 7-bit node IDs in (X, Y, Port, DeviceID) format.

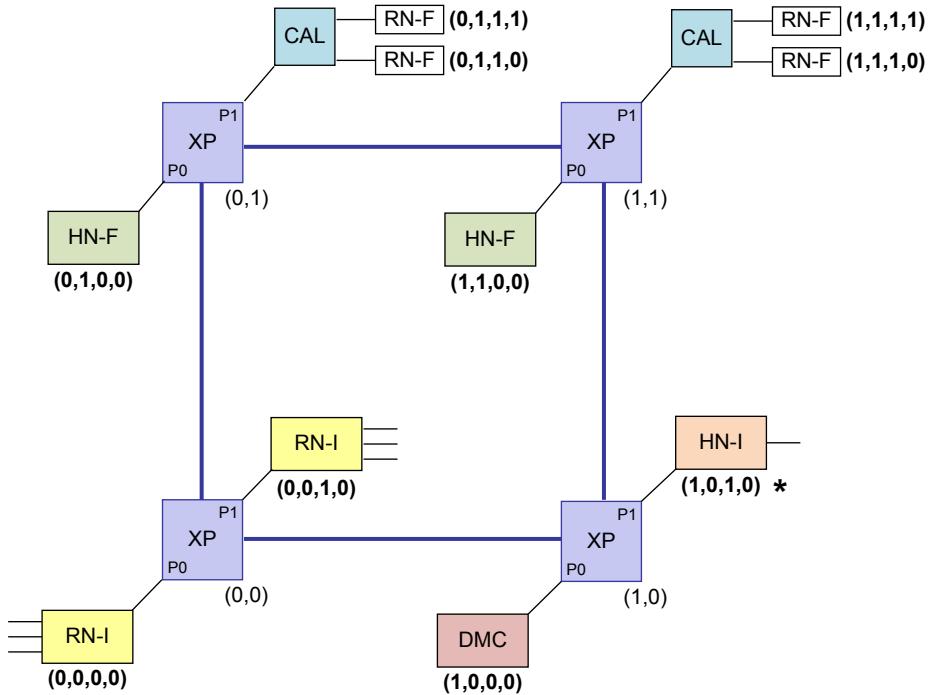


Figure 2-41 Example system with 7-bit node IDs and CAL2

The following figure shows a CMN-700 system with CAL4 and 7-bit node IDs in (X, Y, Port, DeviceID) format.

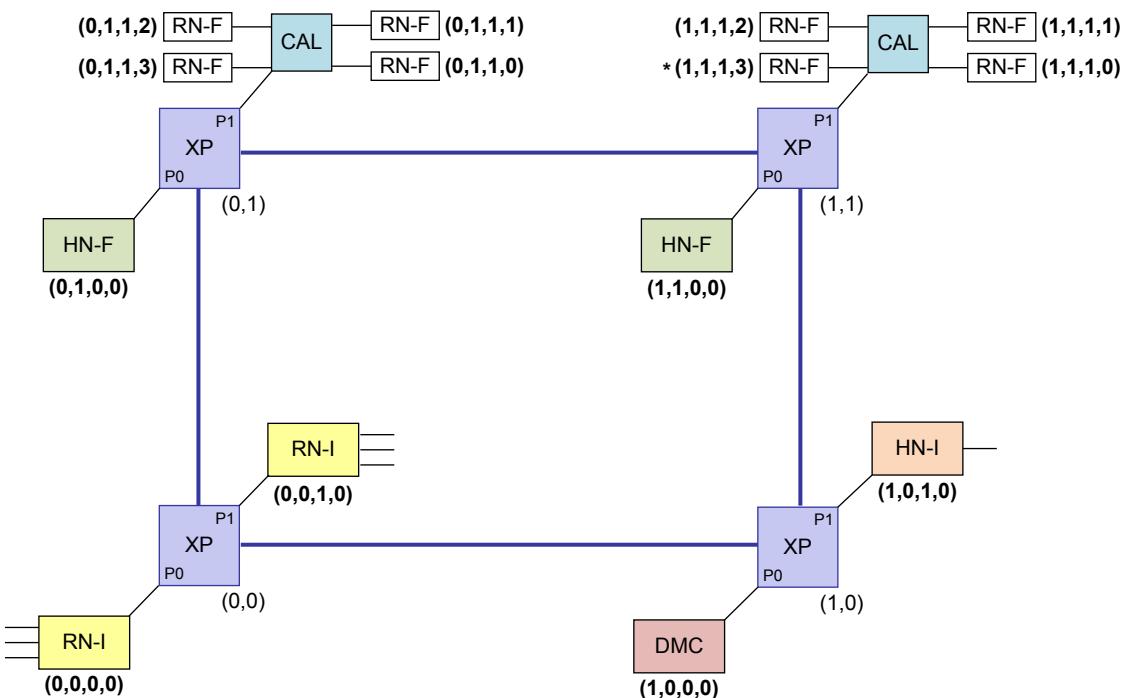


Figure 2-42 Example system with 7-bit node IDs and CAL4

Example 2-3 7-bit node ID format with CAL4

For the third RN-F connected to the CAL on XP (1,1), the node ID reads as (1,1,1,3).

This format is equivalent to 0b01, 0b01, 0b1, 0b11, or 0x2F.

2.4.2 Node ID mapping for configurations with extra device ports

The node ID mapping for a CMN-700 configuration changes from the default scheme when you use extra device ports. In this case, the way that node IDs are mapped depends on whether you are using a mesh configuration or a single-MXP configuration.

The node ID-mapping scheme which is described here is used for configurations with extra device ports. This scheme is different to the default scheme. If your configuration does not use extra device ports, see [2.4.1 Node ID mapping on page 2-95](#).

CHI device node ID for mesh configurations

Note

The naming convention for I/O signals uses decimal values of the node ID. For example, RXREQFLIT_NIDxxx uses xxx values in decimal.

The following items determine the physical position of a device in a CMN-700 mesh configuration with extra device ports:

1. The X coordinate of its XP
2. The Y coordinate of its XP
3. The XP device port (0, 1, 2, or 3) that it connects to

The CHI device node ID is mapped to (X, Y, Device port plus device ID).

The node ID size depends on the X and Y dimensions of the CMN-700 mesh. The larger of the X and Y dimensions determines the size as shown in the following table.

Table 2-20 Node ID size selection

X mesh dimension	Y mesh dimension	Node ID size
4 or less	4 or less	7 bits
5-8	1-2	9 bits
1-2	5-8	

The following tables show the 7-bit and 9-bit node ID format for a CMN-700 mesh configuration using extra device ports.

Table 2-21 7-bit node ID format for mesh configuration with extra device ports

[6:5]	[4:3]	[2:0]
X position	Y position	Device port plus device ID [2:1] Indicates device port [0] Indicates device ID on the device port

Table 2-22 9-bit node ID format for mesh configuration with extra device ports

[8:6]	[5:3]	[2:0]
X position	Y position	Device port plus device ID
	[2:1]	Indicates device port
	[0]	Indicates device ID on the device port

The following figure shows a CMN-700 system with node IDs in (X, Y, Device port plus device ID) format.

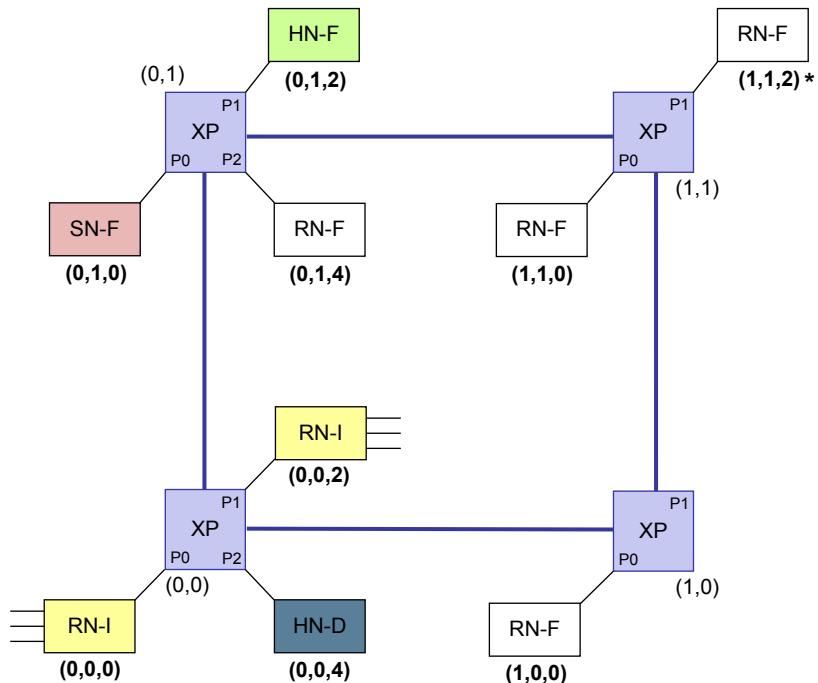


Figure 2-43 Example mesh system with node IDs

Example 2-4 Node ID format for mesh system with extra device ports

For the RN-F on P1 of XP (1,1), the node ID reads as (1,1,2).

This format is equivalent to (0b01, 0b01, 0b010) or 0x2A.

If using CAL in a CMN-700 mesh configuration with extra device ports, only CAL2 configurations are permitted. If CAL is present, the devices that are connected to the CAL are assigned consecutive node IDs. For example, consider two devices that are attached to a CAL on P0. One device is assigned NodeID[2:0]=0b000 and the other device is assigned NodeID[2:0]=0b001.

The following figure shows a CMN-700 system with CAL and node IDs in (X, Y, Device port plus device ID) format.

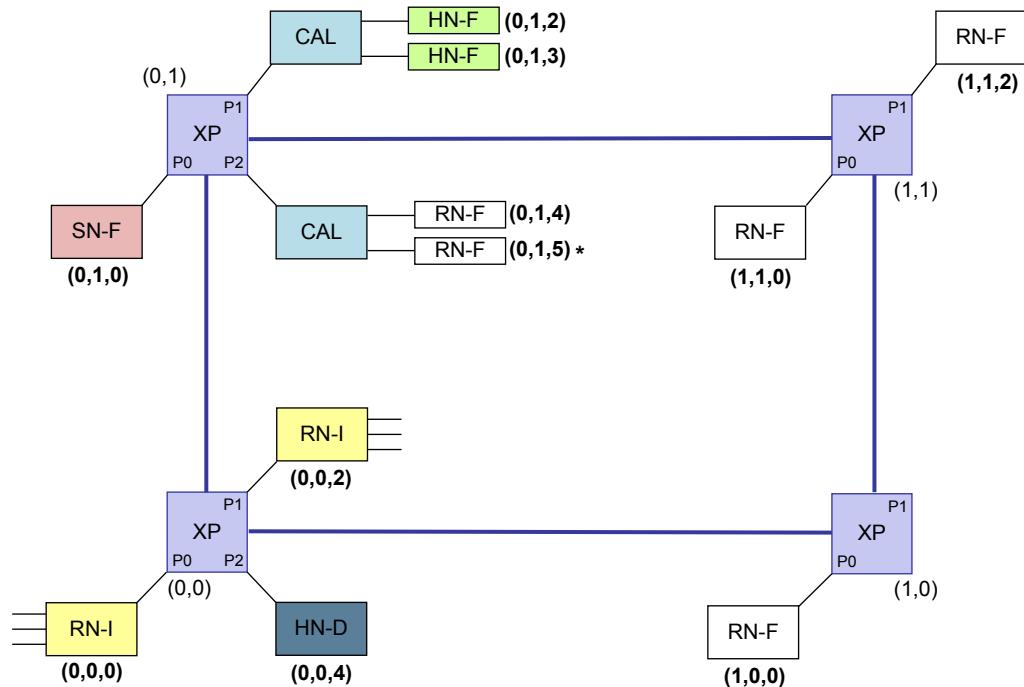


Figure 2-44 Example mesh system using CAL with node IDs

Example 2-5 Node ID format for mesh system with extra device ports and CAL

For the second RN-F attached to the CAL on P2 of XP (0,1), the node ID reads as (0,1,5).

This format is equivalent to (0b00, 0b01, 0b101) or 0xD.

The following table shows the possible node ID[2:0] encodings for a mesh configuration with extra device ports.

Table 2-23 Node ID[2:0] encodings for mesh configuration with extra device ports

Device port number	Device number	Node ID[2:1]	Node ID[0]
P0	0	0b00	0b0
	1	0b00	0b1
P1	0	0b01	0b0
	1	0b01	0b1
P2	0	0b10	0b0
	1	0b10	0b1
P3	0	0b11	0b0
	1	0b11	0b1

CHI node IDs for single-MXP configurations

Which XP device port a device connects to determines the physical location of a device in a CMN-700 single-MXP configuration. For single-MXP configurations, the device port can be P0, P1, P2, P3, P4, or P5.

The device node ID is mapped to (Port, Device).

The following table shows the CHI node ID format for a CMN-700 single-MXP configuration with extra device ports.

Table 2-24 CHI node ID format for single-MXP configuration with extra device ports

[10:5]	[4:2]	[1:0]
Unused	Port	Device

The following figure shows a CMN-700 single-MXP system with node IDs in (Port, Device) format.

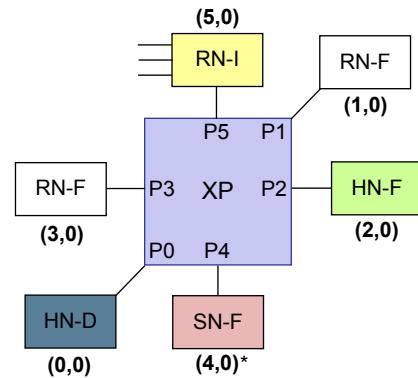


Figure 2-45 Example single-MXP system with node IDs

Example 2-6 Node ID format for single-MXP with extra device ports

For the SN-F attached to P4, the node ID reads as (4,0).

This format is equivalent to (0b100, 0b00) or 0x10.

If using CAL in a CMN-700 single-MXP configuration with extra device ports, all CAL configurations are permitted. If CAL is present, the devices that are connected to the CAL are assigned consecutive node IDs. For example, if two devices are attached to a CAL, one device is assigned NodeID[1:0]=0b00 and the other device is assigned NodeID[1:0]=0b01.

The following figure shows a CMN-700 single-MXP system with CAL and node IDs in (Port, Device) format.

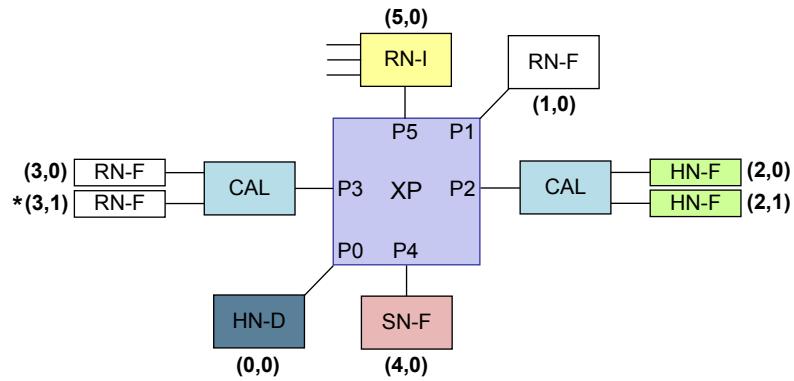


Figure 2-46 Example single-MXP CAL system with node IDs

Example 2-7 Node ID format for single-MXP CAL system

For the second RN-F connected to the CAL on device port P3, the node ID reads as (3,1).

This format is equivalent to (0b011, 0b01) or 0xD.

The following table shows the possible node ID[4:0] encodings for a single-MXP configuration with extra device ports.

Table 2-25 Node ID[4:0] encodings for single-MXP configuration with extra device ports

Target device port ID	Target device ID	Node ID[4:2]	Node ID[1:0]
0	0	0b000	0b00
	1	0b000	0b01
1	0	0b001	0b00
	1	0b001	0b01
2	0	0b010	0b00
	1	0b010	0b01
3	0	0b011	0b00
	1	0b011	0b01
4	0	0b100	0b00
	1	0b100	0b01
5	0	0b101	0b00
	1	0b101	0b01

2.4.3 Addressing capabilities

CMN-700 supports a 34-bit, 44-bit, 48-bit, or 52-bit *Physical Address* (PA) width which defines the PA space for which read and write transactions are supported in the interconnect. The PA width is configured using Socrates and results in the PA_WIDTH global parameter in the CMN-700 RTL.

CHI interfaces in CMN-700 support 44-bit, 48-bit, and 52-bit address field widths for REQ channel flits. This width is also configured using Socrates and results in the REQ_ADDR_WIDTH global parameter in the CMN-700 RTL.

The address field width for SNP channel flits is derived automatically based on the REQ_ADDR_WIDTH global parameter.

The following table shows the legal combinations of physical address widths and flit address widths.

Table 2-26 Legal combinations of physical address and flit address widths

Physical address width	REQ flit address width	SNP flit address width (derived)
34b	44b	41b
34b	48b	45b
44b	44b	41b
44b	48b	45b
48b	48b	45b
52b	52b	49b

The minimum PA width that CCIX supports is 48 bits. If CMN-700 has a PA_WIDTH of less than 48, then in a CML system with non-CMN-700 CCIX components, there is a mismatch between address widths. This mismatch means that the upper address bits of a CCIX request or snoop from a remote CCIX component might be truncated in CMN-700. For example, in this kind of system, bits [47:34] or [47:44] might be truncated in CMN-700. To prevent this truncation, software must ensure that non-CMN-700 CCIX components do not present requests and snoops with addresses higher than the CMN-700 PA_WIDTH to CMN-700.

2.4.4 System Address Map

Every master that is connected to CMN-700 has the same view of memory.

The entire addressable space can be partitioned into subregions, and each partition must be designated as one of the following:

I/O space

HN-I, HN-D, HN-P, HN-T, and HN-V service requests to I/O space.

DDR space

HN-F, SN-F, SBSX, and MTSX service requests to DDR space.

————— Note ————

Unmapped addresses are routed to the HN-D.

Each HN-F covers a mutually exclusive portion of the system address space. The options and constraints for HN-Fs are:

- Each HN-F can contain an SLC.
- HN-Fs can be combined into *System Cache Groups* (SCGs).
- Each HN-F in an SCG must have the same SLC partition size.
- An address hash function determines the target HN-F within an SCG.

All CHI transactions require a target ID to route packets from source to destination. For addressable requests, a *System Address Map* (SAM) determines the target ID. Each node that can generate a CHI addressable request contains a SAM:

RN SAM

Present for all RNs and CXHA nodes. Generates a target ID for requests to HN-F, HN-I, HN-D, HN-P, HN-T, HN-V, MTSX, SBSX, and SN-F.

CXRA SAM

Present in all CXRA nodes. Generates a target ID for requests to CXHA nodes.

HN-F SAM

Present in all HN-Fs. Generates a target ID for requests to SN-F, MTSX and SBSX.

HN-I SAM

Present in all HN-Is. Maps the address of the incoming CHI request to an I/O subregion for ordering purposes.

2.4.5 RN SAM

Transactions from an RN must pass through an RN SAM to generate a CHI target ID.

CMN-700 RN-Fs, RN-Is, and CXHAs use an RN SAM that is internal to the interconnect.

The RN SAM programming sequence is described in [3.4.3 RN SAM and HN-F SAM programming on page 3-1239](#).

Target IDs

This section describes how the target ID is determined.

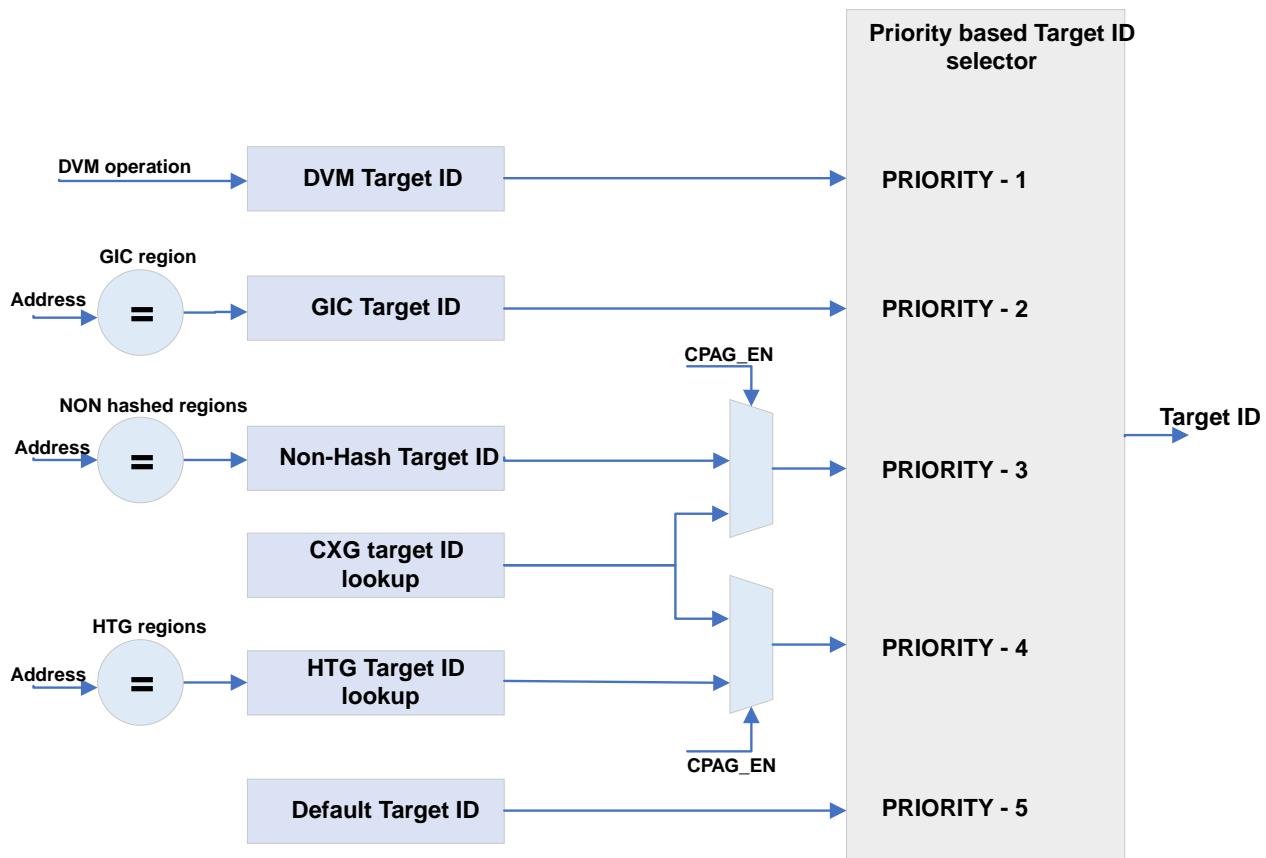


Figure 2-47 RN SAM target ID selection policy

Default target ID

Until the RN SAM has been fully programmed, all addressable transactions use the default target ID. RN SAMs define the HN-D node ID as their default target ID. After the RN SAM has been programmed, the RN SAM only uses the default target ID for addressable requests that do not fall within one of the programmed address regions.

Hashed regions

A given memory partition is distributed (hashed) across many target devices. Hashed Target Regions (HTGs) support up to 32 hashed regions ($RNSAM_NUM_HTG \leq 32$). HTG regions support the following configurations:

- System Cache Groups (SCG): HTG is programmed to support DRAM space to hash across HN-F based on the address
- HN-P and CCG hashing for PCIE traffic: HTG is programmed to support PCIE traffic to hash across HN-P and CCG nodes based on the AXID.

Non-hashed regions

A given memory partition is assigned to an individual targetID (non-hashed). Up to 64 non-hashed regions are supported. The I/O space (HN-I, HN-D, HN-P, HN-V, and HN-T) is intended to be the target of non-hashed regions

Hashed/Non-hashed configuration scenarios

- A hashed region can overlap with a non-hashed region. Whether a given region is configured as hashed or non-hashed affects the target ID selection policy for that address range.
- Using an HTG region register for a single HN-I, HN-D, HN-P, and HN-T target. This scenario might be useful if all the non-hashed region registers have already been used. The region can optionally be classified as a non-hashed region (except for HTG region 0).
- Using a non-hashed region register for an HN-F target. This scenario might be useful if all the HTG region registers have already been used. For target ID selection purposes, this HN-F is classified as a non-hashed region

GIC target ID

RN SAMs also support a *Generic Interrupt Controller* (GIC) memory region which can be used to select GIC-related addresses to a specific target ID. The GIC region can overlap with hashed and non-hashed regions.

RN SAM target ID selection policy

RN SAMs support priority-based target ID selection. The order of priority is when selecting a target ID is:

1. GIC memory region (highest priority)
2. Non-hashed memory region
3. Hashed memory region or non-hashed mode of hashed memory region. These options are mutually exclusive
4. Default memory region (lowest priority)

DVM target ID

DVM transactions are assigned the DVM target ID. RN SAMs define the nodeID of the corresponding Distributed DVM domain target HN-D/HN-V/HN-T as the DVM target ID. DVM target ID is programmable at boot time which overrides the default DVM target ID.

Memory region requirements

This section describes the RN SAM memory region requirements. SAM supports the following two mechanisms to define a memory region, either one of these mechanisms will be enabled in the design through a user parameter.

- Configurable base address & region size ($RNSAM_HTG_RCOMP_EN = 0$) - Legacy CMN mode:

- Each of the programmed region sizes must be a power of two and the partition must be size aligned. The region size can range from 64KB–4PB. For example, a 1GB partition must start at a 1GB-aligned boundary.
- Legacy CMN mode is supported for all the memory regions.
- Minimum size GIC region is 64KB
- Minimum size for other regions (Hashed, non-hashed) is 64MB
- Configurable lower address & upper address (RNSAM-HTG_RCOMP_EN = 1):
 - No restrictions on size of the region
 - Range comparison mode is supported for all memory regions except for GIC region
 - Minimum size for the region is defined using user parameter (RNSAM-HTG_RCOMP_LSB = [16-26])
 - RNSAM-HTG_RCOMP_LSB = 16, defines minimum memory size = 64 KB
 - RNSAM-HTG_RCOMP_LSB = 17, defines minimum memory size = 128 KB
 - RNSAM-HTG_RCOMP_LSB = 20, defines minimum memory size = 1 MB
 - RNSAM-HTG_RCOMP_LSB = 26, defines minimum memory size = 64 MB
- Non-hashed regions have separate parameter to define memory regions:
 - RNSAM_NONHASH_RCOMP_EN
 - RNSAM_NONHASH_RCOMP_LSB

Legacy CMN RN SAM mode supports complex memory maps where DRAM region sizes are not a power of two or are not size aligned. For example, the following figure shows a memory map where the entire address is assigned to a hashed region. Then, non-hashed regions can be individually programmed, because of their higher target ID selection priority. Software must prevent accesses to addresses in a hashed region that are not actually backed by physical memory. For more information, refer to the Principles of Arm® Memory Maps White Paper.

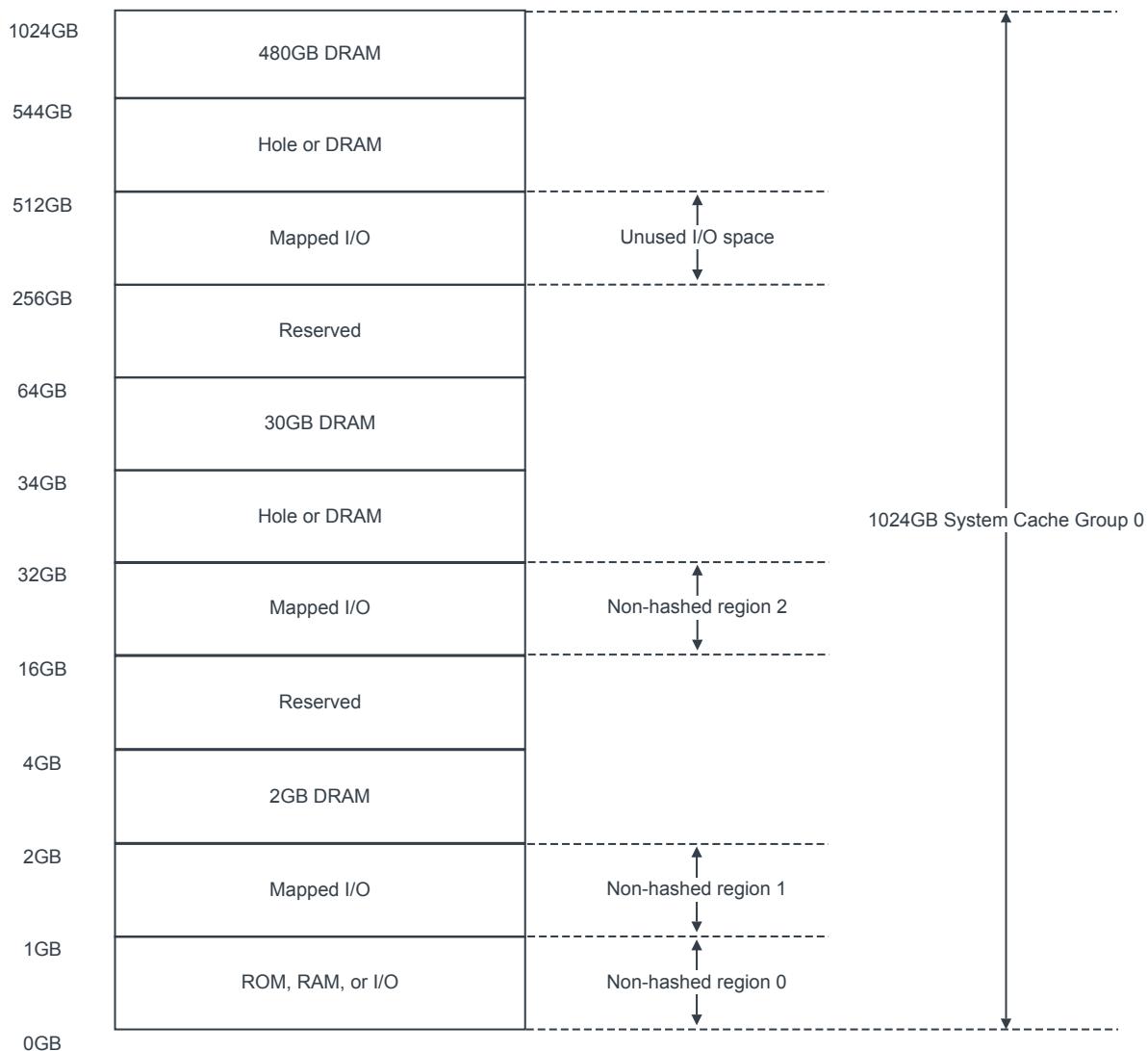


Figure 2-48 Example memory map

For more information on RN SAM configuration register memory partition sizes, see [2.4.8 SAM memory region size configuration](#) on page [2-136](#).

Hashed Target Groups

Hashed Target Groups (HTGs) are configured to support:

- Hashing across HN-F (System Cache Group, address based hashing)
- Hashing across HN-P (PCIE traffic, AXID based hashing)

System Cache Groups

A *System Cache Group* (SCG) is a group of HN-Fs that share a contiguous address region. However, the addresses that are covered by each HN-F in an SCG are mutually exclusive. An HN-F belonging to an SCG is selected as the target based on a hash function.

SCG supports different Hashing mechanisms:

- Power of two hashing (Default, Backward compatible, HN-F nodes = 2/4/8/16/32/64/128/256)
- Non-power of two hashing (HN nodes = [2-255])

- Hierarchical hashing
- User-defined hashing

HTG: Power of two hashing (legacy CMN mode)

An SCG supports hashing over 1, 2, 4, 8, 16, 32, 64, 128, or 256 HN-Fs, using bits[MSB:6] of the PA of the request. If CMN-700 has been configured to implement fewer than 52 PA bits, the unused upper bits are assumed to be zero. The hash algorithm calculates a pointer in the HN-F ID table in the RN SAM. The hash function is explicitly given in the following list. All numbers on the right-hand side of the equations in the list are bit positions within the PA. For example, 17 corresponds to PA bit[17]. In the equations, \wedge represents XOR.

- Two HN-Fs:
 - Number of bits in select: 1
 - select [0] = $(6 \wedge 7 \wedge 8 \wedge \dots \wedge 51)$
- Four HN-Fs:
 - Number of bits in select: 2
 - select [0] = $(6 \wedge 8 \wedge 10 \wedge \dots \wedge 50)$
 - select [1] = $(7 \wedge 9 \wedge 11 \wedge \dots \wedge 51)$
- Eight HN-Fs:
 - Number of bits in select: 3
 - select [0] = $(6 \wedge 9 \wedge 12 \wedge \dots \wedge 51)$
 - select [1] = $(7 \wedge 10 \wedge 13 \wedge \dots \wedge 49)$
 - select [2] = $(8 \wedge 11 \wedge 14 \wedge \dots \wedge 50)$
- 16 HN-Fs:
 - Number of bits in select: 4
 - select [0] = $(6 \wedge 10 \wedge 14 \wedge \dots \wedge 50)$
 - select [1] = $(7 \wedge 11 \wedge 15 \wedge \dots \wedge 51)$
 - select [2] = $(8 \wedge 12 \wedge 16 \wedge \dots \wedge 48)$
 - select [3] = $(9 \wedge 13 \wedge 17 \wedge \dots \wedge 49)$
- 32 HN-Fs:
 - Number of bits in select: 5
 - select [0] = $(6 \wedge 11 \wedge 16 \wedge \dots \wedge 51)$
 - select [1] = $(7 \wedge 12 \wedge 17 \wedge \dots \wedge 47)$
 - select [2] = $(8 \wedge 13 \wedge 18 \wedge \dots \wedge 48)$
 - select [3] = $(9 \wedge 14 \wedge 19 \wedge \dots \wedge 49)$
 - select [4] = $(10 \wedge 15 \wedge 20 \wedge \dots \wedge 50)$
- 64 HN-Fs:
 - Number of bits in select: 6
 - select [0] = $(6 \wedge 12 \wedge 18 \wedge \dots \wedge 48)$
 - select [1] = $(7 \wedge 13 \wedge 19 \wedge \dots \wedge 49)$
 - select [2] = $(8 \wedge 14 \wedge 20 \wedge \dots \wedge 50)$
 - select [3] = $(9 \wedge 15 \wedge 21 \wedge \dots \wedge 51)$
 - select [4] = $(10 \wedge 16 \wedge 22 \wedge \dots \wedge 46)$
 - select [5] = $(11 \wedge 17 \wedge 23 \wedge \dots \wedge 47)$
- 128 HN-Fs
 - Number of bits in select: 7
 - select [0] = $(6 \wedge 13 \wedge 20 \wedge \dots \wedge 48)$
 - select [1] = $(7 \wedge 14 \wedge 21 \wedge \dots \wedge 49)$
 - select [2] = $(8 \wedge 15 \wedge 22 \wedge \dots \wedge 50)$
 - select [3] = $(9 \wedge 16 \wedge 23 \wedge \dots \wedge 51)$
 - select [4] = $(10 \wedge 17 \wedge 24 \wedge \dots \wedge 45)$
 - select [5] = $(11 \wedge 18 \wedge 25 \wedge \dots \wedge 46)$
 - select [6] = $(12 \wedge 19 \wedge 26 \wedge \dots \wedge 47)$
- 256 HN-Fs:
 - Number of bits in select: 8
 - select [0] = $(6 \wedge 14 \wedge 22 \wedge \dots \wedge 46)$

- select [1] = (7^15^23^...^47)
- select [2] = (8^16^24^...^48)
- select [3] = (9^17^25^...^49)
- select [4] = (10^18^26^...^50)
- select [5] = (11^19^27^...^51)
- select [6] = (12^20^28^...^44)
- select [7] = (13^21^29^...^45)

SCG: Non-power of two hashing

In this mode, TargetID index is computed through 12-bit hash followed by modulo operation. In this hashing mode, DRAM memory distribution across HN-F are non-uniform. This hashing type is enabled through user parameter (RNSAM_NP2_EN).

SCG: Hierarchical hashing

HN-Fs are grouped hierarchically, the first hierarchy is a cluster of HN-Fs, and the second hierarchy are the HN-Fs within a cluster. The number of clusters are constrained to be power of two (supported clusters 2,4,8,16,32).

Number of cluster and number of HN-Fs per cluster are boot time configurable. The second hierarchy supports non-power of two hashing to the HN-Fs within the clusters, and maximum number of HN-Fs per cluster is 32. This hashing type is enabled through user parameter (RNSAM_HIER_HASH_EN).

In Hierarchical hashing, memory address hashed at first hierarchy is routed to only a subset of HNF's in the second hierarchy and hence memory controllers downstream to HN-F's should be configured to support either direct mapped per cluster, direct mapped per HN-F or direct mapped per SCG

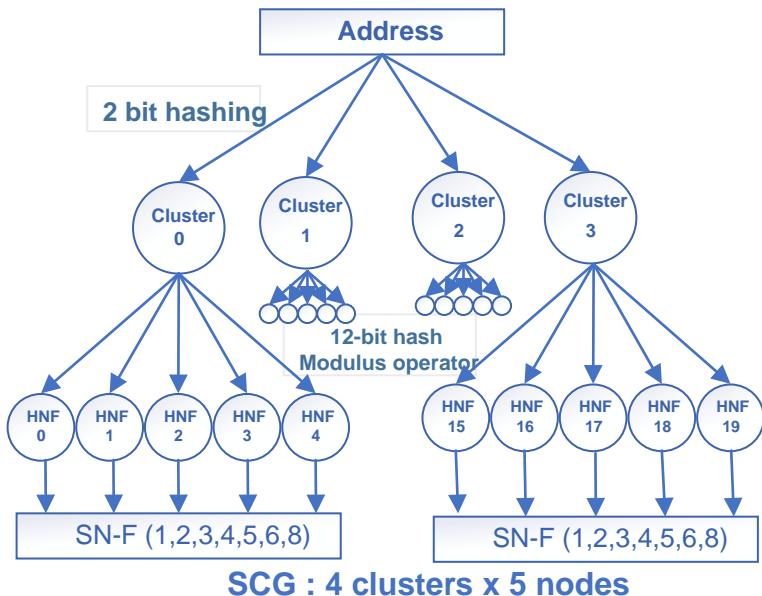


Figure 2-49 Hierarchical hashing configuration

SCG: User-defined hashing logic

The RTL hash modules can be selectively replaced to implement user defined hash logic. RN SAM implements user-defined configuration registers to be used for the user-defined hash logic. The number of user defined registers are enabled using a user parameter (RNSAM_CUSTOM_REGS [0-8]). See the *Arm® Neoverse™ CMN-700 Coherent Mesh Network Configuration and Integration Manual* for more information on modifying the RTL hash modules.

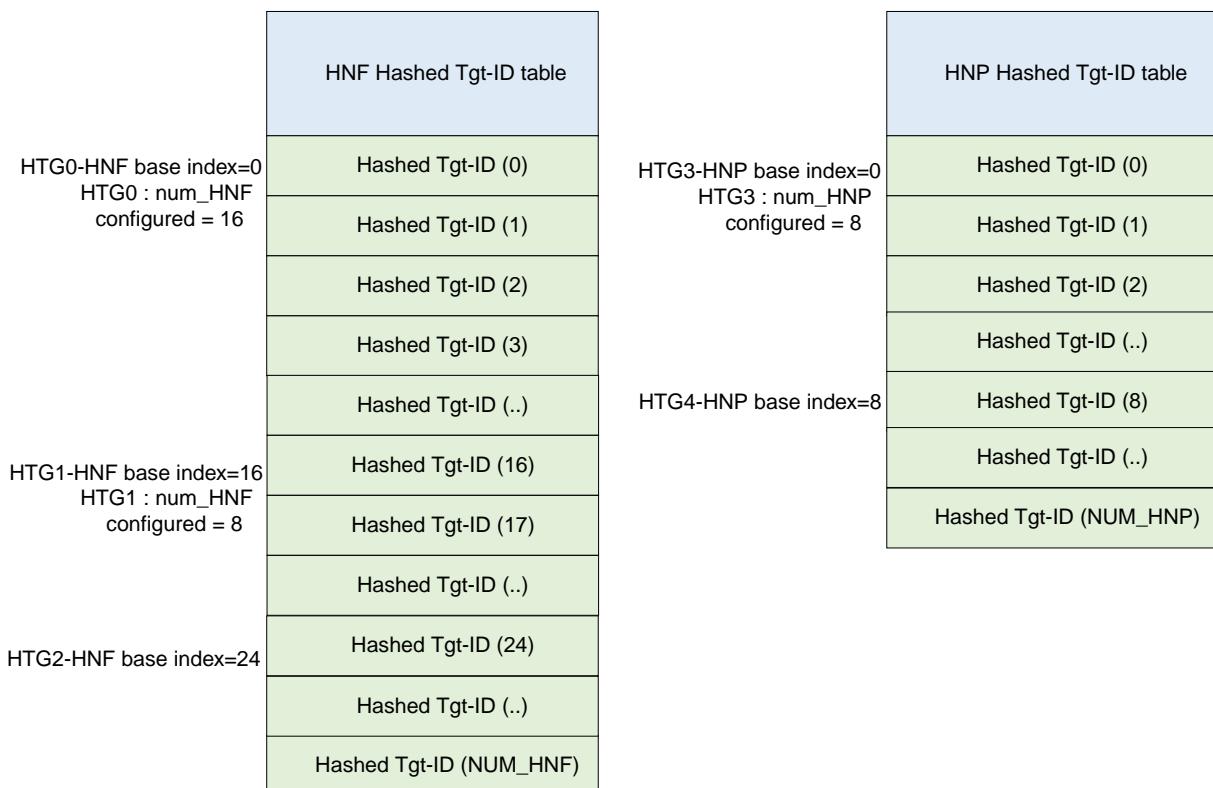
SCG configuration

CMN-700 SCGs occupy the lower four HTGs. This restriction is placed to preserve the backward compatibility. HTG supports maximum of 128 HN-F Target IDs (256 total HN-Fs with CAL2) which are either configured for one SCG or distributed across SCGs

Each HTG is associated with a set of HN-F Target IDs defined through the Target ID table base index for each HTG. HN-F target IDs are associated to SCGs in the following two mechanisms.

- Flexible Target ID base indexes (RNSAM_FLEX_TGTID_EN = 1). At Reset all the HTG base indexes point to the zero. Based on the programming, base indexes of each HTG are derived like a linked list approach.

$$\text{HTG}[n] \text{ base index} = \text{HTG}[n-1] \text{ base index} + \text{HTG } [n-1] \text{ num HNF}$$



- Hardcoded base index for each SCG (Legacy CMN mode, RNSAM_FLEX_TGTID_EN = 0). This Legacy CMN mode is only applicable for (HTG = SCG = 4, NUM_HNF <= 64).

For CMN-700 configuration of SCG, The following table shows the restrictions on SCG selection when there are 16, 32, or 64 HN-Fs in a given SCG.

Table 2-27 Permitted allocation of HN-Fs into SCGs

SysCacheGroup HN-F/SN-F counts.	1 HN-F	2 HN-F	4 HN-F	8 HN-F	16 HN-F	32 HN-F	64 HN-F
SysCacheGroup 0	Y	Y	Y	Y	Y	Y	Y
SysCacheGroup 2	Y	Y	Y	Y	Y	Y	N
SysCacheGroup 1	Y	Y	Y	Y	Y	N	N
SysCacheGroup 3	Y	Y	Y	Y	Y	N	N

The RN SAM supports up to 64 hashed HN-F and SN-F target IDs without using CAL mode. This feature allows up to 64 unique hashed target IDs in the RN SAM SCG target nodeID registers. RN SAM also supports up to 32 hashed target IDs when using CAL mode.

The nodeIDs in sys_cache_grp_[hn, sn]_nodeid_reg<X> registers are shared between all the SCGs. Therefore, the number of nodeIDs that are available for each SCG depends on the number of HN-Fs or CALs. The following algorithm determines the distribution of nodeIDs.

SCG0	NodeID0 to nodeID[n - 1]
SCG1	NodeID[n / 4] to nodeID[(n / 2) - 1]
SCG2	NodeID[n / 2] to nodeID[n - 1]
SCG3	NodeID[n x 3 / 4] to nodeID[n - 1]

In the preceding algorithm, n represents the total number of hashed target IDs in the SAM.

If SCG0 uses all the available nodeIDs, then SCG1, SCG2, and SCG3 must not be used. If SCG0 only uses nodeID0 through nodeID[(n / 2) - 1], then SCG1 cannot be used. However, in this case, you can use SCG2 and SCG3 with (n / 4) nodeIDs in each of the SCGs.

For example, the following table shows the register and nodeID allocation for each SCG in a system with 64 hashed target IDs.

Table 2-28 RN SAM SCG target ID programming for 64 hashed targets

SCG target ID registers (64 hashed targets)	Number of HN-Fs per SCG target ID table				
	64	32	16	8	4,2,1
SCG CAL mode supported	No	Yes	Yes	Yes	Yes
sys_cache_grp_hn_nodeid_reg0	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs
sys_cache_grp_hn_nodeid_reg1				-	
sys_cache_grp_hn_nodeid_reg2			-	-	
sys_cache_grp_hn_nodeid_reg3			-	-	
sys_cache_grp_hn_nodeid_reg4		SCG1_NIDs	SCG1_NIDs	SCG1_NIDs	SCG1_NIDs
sys_cache_grp_hn_nodeid_reg5				-	
sys_cache_grp_hn_nodeid_reg6			-	-	
sys_cache_grp_hn_nodeid_reg7			-	-	
sys_cache_grp_hn_nodeid_reg8		SCG2_NIDs	SCG2_NIDs	SCG2_NIDs	SCG2_NIDs
sys_cache_grp_hn_nodeid_reg9				-	
sys_cache_grp_hn_nodeid_reg10			-	-	
sys_cache_grp_hn_nodeid_reg11			-	-	
sys_cache_grp_hn_nodeid_reg12		SCG3_NIDs	SCG3_NIDs	SCG3_NIDs	SCG3_NIDs
sys_cache_grp_hn_nodeid_reg13				-	
sys_cache_grp_hn_nodeid_reg14			-	-	
sys_cache_grp_hn_nodeid_reg15			-	-	

The following table shows the register and nodeID allocation for each SCG in a system with 16 hashed target IDs.

Table 2-29 RN SAM SCG target ID programming for 16 hashed targets

SCG target ID registers (16 hashed targets)	Number of HN-Fs per SCG target ID table					
	16	8	4	2, 1		
sys_cache_grp_hn_nodeid_reg0	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs		
				-		
		SCG1_NIDs	SCG1_NIDs	SCG1_NIDs		
				-		
sys_cache_grp_hn_nodeid_reg1	SCG2_NIDs	SCG2_NIDs	SCG2_NIDs	SCG2_NIDs		
				-		
		SCG3_NIDs	SCG3_NIDs	SCG3_NIDs		
sys_cache_grp_hn_nodeid_reg2	SCG3_NIDs			SCG3_NIDs		
sys_cache_grp_hn_nodeid_reg3				SCG3_NIDs		

The following table shows the register and nodeID allocation for each SCG in a system with eight hashed target IDs.

Table 2-30 RN SAM SCG target ID programming for eight hashed targets

SCG target ID registers (eight hashed targets)	Number of HN-Fs per SCG target ID table			
	8	4	2	1
sys_cache_grp_hn_nodeid_reg0	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs
				-
		SCG1_NIDs	SCG1_NIDs	SCG1_NIDs
				-
sys_cache_grp_hn_nodeid_reg1	SCG2_NIDs	SCG2_NIDs	SCG2_NIDs	SCG2_NIDs
				-
		SCG3_NIDs	SCG3_NIDs	SCG3_NIDs
				-

The following table shows the register and nodeID allocation for each SCG in a system with four hashed target IDs.

Table 2-31 RN SAM SCG target ID programming for four hashed targets

SCG target ID registers (four hashed targets)	Number of HN-Fs per SCG target ID table		
	4	2	1
sys_cache_grp_hn_nodeid_reg0	SCG0_NIDs	SCG0_NIDs	SCG0_NIDs
			SCG1_NIDs
		SCG2_NIDs	SCG2_NIDs
			SCG3_NIDs

The hashed target ID allocation in the preceding tables is also applicable to SN target IDs.

The following table shows an example mapping of HN-Fs to SCGs.

Table 2-32 25 HN-Fs to three SCGs programming example

SCG number	Number of HN-Fs	Node ID
SCG0	16	NID0-15
SCG2	8	NID16-23
SCG3	1	NID24

The following table shows example programming for 25 HN-Fs.

Table 2-33 Example programming for 25 HN-Fs

SCG target ID registers.	Number of HN-Fs per SCG			
	32	16	8	1
sys_cache_grp_hn_nodeid_reg0	-	SCG0	-	-
sys_cache_grp_hn_nodeid_reg1		NID0-15		
sys_cache_grp_hn_nodeid_reg2				
sys_cache_grp_hn_nodeid_reg3				
sys_cache_grp_hn_nodeid_reg4		-	SCG2	
sys_cache_grp_hn_nodeid_reg5			NID16-23	
sys_cache_grp_hn_nodeid_reg6		-		SCG3 NID24
sys_cache_grp_hn_nodeid_reg7				-

SCGs 1-3 can be configured to non-hashed mode. In non-hashed mode, the SCG can contain a single HN-I, HN-T, HN-D, or HN-F.

Hashing across HN-P and CCG (PCIe traffic, AXID based hashing)

In this mode, the incoming PCIe traffic is hashed across multiple HN-P based on the AXID, that is the ARID or AWID of the request presented to the RN-I or RN-D ACE-Lite interface . HN-P target IDs associated to each HTG are derived through Flexible base indexes similar to HN-F target IDs, legacy CMN mode of hardcoded base indexes is not supported for AXID based hashing.

rnsam_axid_in_port = {port_id, AXID} = {2bits, 32 bits} = 34 bit. AXID interface is zero extended when AXID interface is less than 32 bits. This hashing type is enabled through user parameter RNSAM_AXID_HASH_EN and RNSAM_FLEX_TGID_EN.

An HTG supports hashing over 1, 2, 4, 8, 16, 32 HN-Ps/CCGs, using the full AXID of the request and AXI port identifier. The hash algorithm calculates a pointer in the HN-P ID table in the RN SAM. The hash function is explicitly given in the following list:

- Two HN-Ps:
 - Number of bits in select: 1
 - select [0] = $(0^1^2^{\dots}^33)$
- Four HN-Ps:
 - Number of bits in select: 2
 - select [0] = $(0^2^4^{\dots}^32)$
 - select [1] = $(1^3^5^{\dots}^33)$
- Eight HN-Ps:
 - Number of bits in select: 3
 - select [0] = $(0^3^6^{\dots}^33)$
 - select [1] = $(1^4^7^{\dots}^31)$
 - select [2] = $(2^5^8^{\dots}^32)$

- 16 HN-Ps:
 - Number of bits in select: 4
 - select [0] = (0^4^8^...^32)
 - select [1] = (1^5^9^...^33)
 - select [2] = (2^6^10^...^31)
 - select [3] = (3^7^11^...^32)
- 32 HN-Fs:
 - Number of bits in select: 5
 - select [0] = (0^5^10^...^30)
 - select [1] = (1^6^11^...^31)
 - select [2] = (2^7^12^...^32)
 - select [3] = (3^8^13^...^33)
 - select [4] = (4^9^14^...^29)

Secondary Memory Regions for Hashed Target Groups

Each *Hashed Target Groups* supports two additional memory regions. If an incoming address match either of the two programmed, valid regions, the RN SAM selects the corresponding target ID of the HTG from the target ID table.

The following limitations apply:

- If the primary region for an SCG is set to be in non-hashed mode, the secondary region is also set to be in non-hashed mode.

Address range-based QoS override and PrefetchTgt support

CMN-700 provides resources that can override the QoS value of requests and facilitate PrefetchTgt operations to specific address ranges.

The value of the RNSAM_NUM_QOS_REGIONS configuration parameter and the value of a bit within the sam_qos_mem_region_reg* registers determine which features are enabled. The following table shows the possible modes and their respective values.

Table 2-34 QoS override and PrefetchTgt mode configuration values

RNSAM_NUM_QOS_REGIONS value	sn_tgtid_override value	
0	0	1
0-7	QoS override mode only enabled	
8	QoS override mode only enabled	QoS override mode and PrefetchTgt mode enabled

The QoS override mode and the PrefetchTgt mode are described in the following sections.

Address range-based QoS override

You can program the RN SAM to override the QoS value of requests from RNs to HNs that target certain memory regions. This feature is present in all instances of the RN SAM inside MXP, RN-I, RN-D, and CXHA.

You can use this feature to change the priority of traffic targeting high-priority or low-priority memory or I/O devices. With this feature, requests that pass through the same QoS regulators can have different QoS values.

You configure the number of override memory regions using the RNSAM_NUM_QOS_REGIONS parameter. Each region corresponds to one of the sam_qos_mem_region_reg* registers. Each of these registers contains a bit to indicate the following details for the region:

- Region valid: indicates that the programmed memory region is valid for comparison
- Region base address
- Region size

- Region QoS value
- Override bit: determines whether override occurs for that memory region or not

The QoS regions follow the same base address and size properties as hashed and non-hashed regions.

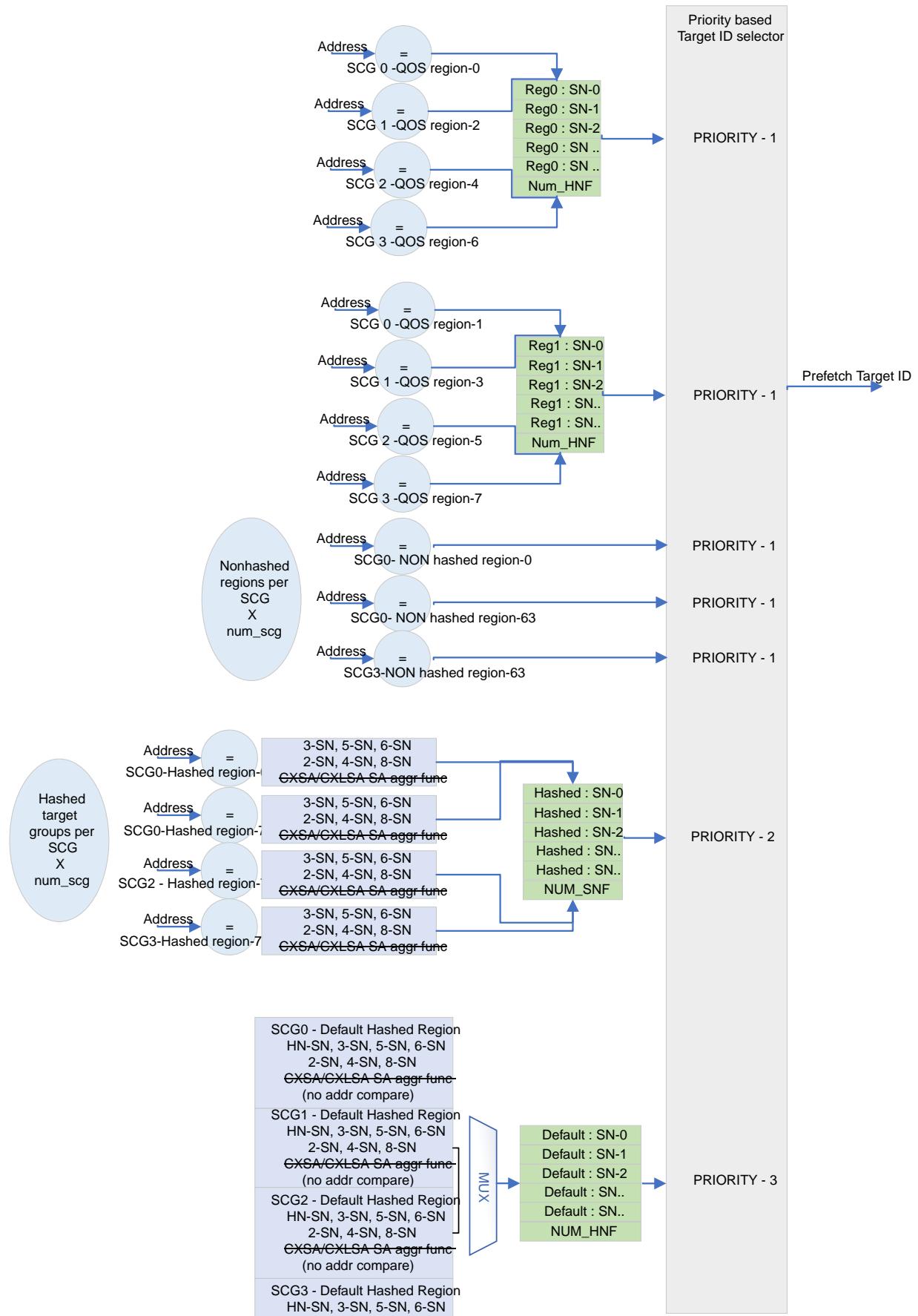
When you enable the Address range-based QoS override feature, CMN-700 compares the incoming address against the valid QoS memory regions. If the address matches any of the programmed valid addresses, the request flit uses the corresponding QoS override value.

The memory regions that you specify for override are independent of the hashed and non-hashed memory regions in the RN SAM. Therefore, QoS override regions can overlap with either of the hashed or non-hashed regions. However, two QoS override regions must not overlap with each other.

The RN SAM QoS memory regions and QoS override do not apply to the GIC memory region. Therefore, you must not specify the GIC memory region as a target for QoS override.

Support for PrefetchTgt operations in RN SAM

The RN SAM supports CHI PrefetchTgt operations. These operations are sent from RN-F directly to SN-F, bypassing the HN-F. The RN SAM prefetchTgt target ID generation must comprehend both the RN SAM HN-F target ID generation and HN-F SAM SN target ID generation.



To support PrefetchTgt operations, the RN SAM integrates the functionality of HN-F SAM to determine the appropriate SN-F target ID for a given address. RN-Fs that integrate the PrefetchTgt RN SAM only use the SN-F target ID for PrefetchTgt requests. The PrefetchTgt RN SAM programming must match the HN-F SAM for SN-F target IDs. RNSAM supports PrefetchTgt IDs for a maximum of 8 SCGs, RNSAM_PFTGT_NUM_SCG parameter.

The PrefetchTgt is configured to match the HN-F SAM configuration for the associated address range:

- Direct-mapped or address-striped in the HN-F SAM (Default Hashed Region)
- Non-hashed target regions in the HN-F
- HTG region in the HN-F

PrefetchTgt operations to direct-mapped or address-striped SN-F targets

The following registers are used to program the PrefetchTgt functionality in the RN SAM:

- por_rnsam_sys_cache_grp_sn_attr
- por_rnsam_sys_cache_grp_sn_nodeid_reg{0-7}
- por_rnsam_sys_cache_grp_sn_sam_cfg{0-1}

The PrefetchTgt RN SAM registers are only present in RN SAM blocks associated with CHI-B, CHI-C, CHI-D, or CHI-E RN-F nodes.

————— Note ————

For RN SAM blocks associated with other node types such as RN-I, RN-D, and CXHA:

- Reads of these register offsets always return a value of zero.
- Writes to these register offsets have no effect.
- These registers do not appear in the IP-XACT files.

This mode is enabled through setting RNSAM_PFTGT_DEF_HASHED_GRP_EN.

Support for PrefetchTgt operations to address range-based targets

The RNSAM_PFTGT_DEF_HASHED_GRP_EN & RNSAM_NUM_QOS_REG==8 parameter enables the incoming address matches one of the specified QoS regions when the sn_tgtid_override bit for the region is set to 1. If the address of the request is within the corresponding SCG, the RN SAM generates the SN target ID as programmed in sys_cache_grp_region[0-1]_sn_nodeid_reg[0-31] registers.

The node IDs of the SNs in these registers are mapped to each SCG similarly to the node IDs of the HNs. The following table shows the mapping of each SCG to the relevant QoS region registers.

————— Note ————

The memory regions that are programmed in the sam_qos_mem_region_regX registers must match the memory regions that are programmed in the por_hnf_sam_memregion[0-1] registers.

Table 2-35 Mapping of SCGs to QoS region registers

SCG ID	Mapped registers
SCG0	<ul style="list-style-type: none"> • sam_qos_mem_region_reg0 • sam_qos_mem_region_reg1
SCG1	<ul style="list-style-type: none"> • sam_qos_mem_region_reg2 • sam_qos_mem_region_reg3

Table 2-35 Mapping of SCGs to QoS region registers (continued)

SCG ID	Mapped registers
SCG2	<ul style="list-style-type: none"> • sam_qos_mem_region_reg4 • sam_qos_mem_region_reg5
SCG3	<ul style="list-style-type: none"> • sam_qos_mem_region_reg6 • sam_qos_mem_region_reg7

PrefetchTgt operations based on HN-F Non-hashed regions

PrefetchTgt operations for the HN-F non-hashed memory regions are supported and assigned a unique SN-F target ID for each non-hashed region. Prefetch non-hash regions also support QOS override capability. RNSAM supports a maximum of 64 non-hashed regions per system cache group (RNSAM_PFTGT_NUM_NONHASH_PSCG). RNSAM supports a total prefetch non-hashed region:

RNSAM_PFTGT_NUM_SCG x RNSAM_PFTGT_NUM_NONHASH_PSCG x

PrefetchTgt operations based on HN-F HTG regions

PrefetchTgt operations for the HN-F HTG regions are supported and assigned a SN-F through address striped hash functions. RNSAM supports a maximum of 8 hashed target regions per system cache group. RNSAM supports a total prefetch hashed target region:

RNSAM_PFTGT_NUM_SCG x RNSAM_PFTGT_NUM_HTG_PSCG

Prefetch Hashed regions supports the following hash functions (MOD-3/5/6 hashing, Power of two hashing (2SN, 4SN, 8SN). More information on these hashing mechanisms are explained in HN-SAM.

Prefetch hashed/non-hashed memory regions support two different ways of defining a memory region:

- Configurable base address & region size (HNSAM_RCOMP_EN = 0) – Legacy CMN mode.
 - Each of the programmed region sizes must be a power of two and the partition must be size aligned. The region size can range from 64KB–4PB. For example, a 1GB partition must start at a 1GB-aligned boundary
 - Legacy CMN mode is supported for all the memory regions.
 - Minimum size for other regions (HAsked, non-hashed) is 64MB.
- Configurable lower address & upper address.
 - No restrictions on size of the region
 - Minimum size for the region is defined using user parameter (HNSAM_RCOMP_LSB = [20-26]).
 - HNSAM_RCOMP_LSB = 20, defines minimum memory size = 1 MB
 - HNSAM_RCOMP_LSB = 26, defines minimum memory size = 64 MB

SN-TgtID lookup for the PrefetchTgt operations

Prefetch operations targeting Hashed regions, Default hashed, and QOS regions derive the SN target ID from the configured SN-Target ID tables.

HTG Regions

Prefetch operations targeting the HN-F HTG regions derive the SN target ID based on the address hashing mechanism, and the resultant hash index is looked up into the SN Tgt-ID table. Each of the prefetch HTG regions has an index to SN-target ID table.

Default hashed region

Prefetch operations targeting the Default hashed regions derive the SN target ID based on the address hashing mechanism or HN-F affinity, and the resultant hash index is looked up into Default hashed SN Tgt-ID table. The number of entries in the table is enabled through the build time parameter (RNSAM_NUM_HNF_TGT). SN-F base indexes for each system cache group follows the same HN-F base index for the associated system cache group.

QoS regions

Prefetch operations targeting the QOS region derive the SN target ID from the QOS region SN Tgt-ID tables. QOS regions-0/2/4/6/8/10/12/14 target the Region0-SN Tgt-ID table, and QOS regions-1/3/5/7/9/11/13/15 target the Region1-SN Tgt-ID table. SN target ID index from these tables follow the HN-F indexes of associated system cache groups.

PrefetchTgt operations to address range-based SN-F targets

The sam_qos_mem_region_reg* registers can be used to send PrefetchTgt operations to range-based targets in the HN-F SAM. These registers are also used to override the QoS value of requests from RN to HN. For information about enabling PrefetchTgt operations to range-based targets and the relationship with the QoS override feature, see [Address range-based QoS override and PrefetchTgt support on page 2-115](#).

HN-F with CAL support

CMN-700 system supports pairing of two HN-Fs at an MXP port using CAL.

The HN-F NodeIDs paired at the CAL are only differentiated using the device ID (NodeID[1:0]) field in the node ID as [2.4.1 Node ID mapping on page 2-95](#) describes.

There are two options for assigning HN-F nodes to System Cache Groups (SCGs) when CALs are used:

Note

The following description of these options uses an example configuration, which is:

- Four CAL instances are used to connect eight HN-F nodes (two HN-F nodes per CAL)
 - All eight HN-F nodes belong to the same SCG
-

Normal mode

When using normal mode, each HN-F node ID is explicitly assigned to an SCG using the methods that [Hashed Target Groups on page 2-108](#) describes and in the previous programming examples. In the preceding example configuration, all eight HN-F node IDs would be entered in the target ID registers of the SCG.

The HN-F count field for that SCG would be set to eight. This approach allows up to 64 HN-F nodes that are connected to 32 CAL instances to be assigned to an SCG.

CAL mode

In CAL mode, only one of the two HN-Fs on the CAL has its node ID entered into the SCG target ID registers. In the preceding example configuration, only four HN-F node IDs would be entered in the SCG target ID registers (one per CAL). The HN-F count field for that SCG would be set to four. This approach allows up to 64 HN-F nodes that are connected to 32 CAL instances to be assigned to an SCG.

Mixed mode

In systems with mixed CAL and non-CAL HN-Fs, a single SCG can contain mix of CAL and non-CAL HN-Fs. The CAL HN-Fs must be individually programmed in normal mode. You must not enable the scg<X>.hnf_cal_mode_en field in the sys_cache_grp_cal_mode_reg register for this kind of SCG.

The default Device ID selection in CAL mode is determined by the LSB of the hash function. The RN SAM allows hash MSB selection for power-of-2 hashing modes. Non-power-of-2 and Hierarchical hashing modes must use the hash LSB to determine Device ID selection.

Table 2-36

Number of HNF target ID's (configured per HTG)	CAL disabled (total HN's hashed)	CAL 2 Enabled (total HN's hashed)
2	2	4
4	4	8
8	8	16
16	16	32
32	32	64
64	64	128
128	128	256

— Note —

If there are HN-F CAL instances that are connected in normal mode in your configuration, it might be necessary to modify the RNSAM_NUM_ADD_HASHED_TGT global RTL parameter when configuring the mesh in Socrates. By default, the number of HN-F CAL instances determines the number of sys_cache_grp_hn_nodeid registers that are rendered in the HN-F, not the number of HN-F nodes. For the number of these registers to at least match the absolute number of HN-F nodes, you must increase the RNSAM_NUM_ADD_HASHED_TGT parameter value by the number of HN-F CAL instances that are present in the mesh.

The RN SAM sys_cache_grp_cal_mode_reg register contains the CAL mode enable bit for each system cache group. For example, to enable the CAL_Mode for System Cache Group 0, write 0b1 to bit[0] of this register.

The RN SAM depends on this scheme to support hashing of addresses over twice the number of HN-Fs using the existing hashed target ID programming table.

For example, consider an SCG that is programmed to have four HN-Fs and HN-F CAL mode enabled for this region. For this SCG, the RN SAM generates eight unique target IDs according to the following function:

Number of bits in select: 3

```
select[0] = (6^9^12^15^18^21^24^27^30^33^36^39^42^45^48^51)
select[1] = (7^10^13^16^19^22^25^28^31^34^37^40^43^46^49)
select[2] = (8^11^14^17^20^23^26^29^32^35^38^41^44^47^50)
```

Bits select[1:0] are used to pick between the programmed four HN-F target IDs. Bit select[2] is used to override the “device ID” field as follows:

Target NodeID[10:0] = {hash_nodeID_pick[10:1], hash_nodeID_pick[0] ^ select[2]}.

The following limitations apply:

- This feature must only be used when the target HN-F are paired using CAL
- RN SAM does not apply this method to SN-F target IDs. To fully utilize CHI-B PrefetchTarget operations to SN-F, the paired HN-Fs must always be mapped to same SN-F or group of SN-Fs if 3-SN, 5-SN, or 6-SN hashing is used.
- Only one HN-F ID from each CAL group must be programmed in the RN SAM hashed target ID registers
- If an SCG contains a mix of local HN-F, CCG and CXG NodeIDs, HN-F CAL mode must not be used for that SCG

Non-hashed region options in RN SAM

The RN SAM supports 8, 12, 16, or 20 unique non-hashed regions.

————— Note ————

For RN-F ESAM node types, there is an extra cycle of latency for RN SAM lookup in the XP on the REQ flit when 12, 16, or 20 non-hashed regions are present.

SAM support for CCIX Port Aggregation

RN SAM and HN-F SAM both support *CML Port Aggregation* (CPA) functionality.

To determine whether CPA is enabled, RN SAM uses the address ranges and HN-F SAM uses the logical ID of the RN-F.

Support for CPA in RN SAM

Requests from an RN to a remote chip can be striped across CXGs or CCGs according to PA as *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0* describes. The following figure shows this functionality.

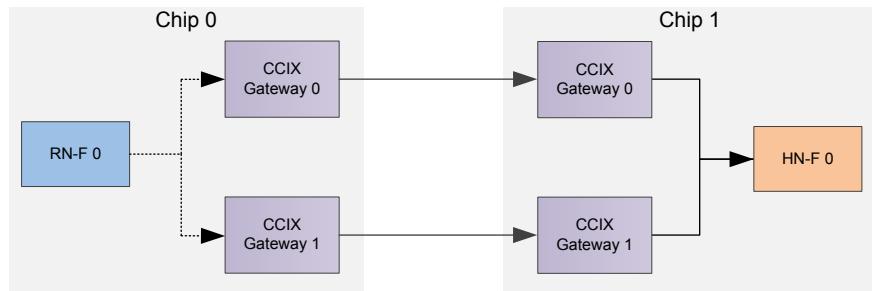


Figure 2-51 RN SAM CCIX Port Aggregation

This striping is achieved by hashing physical address bits[51:6]. The RN SAM can hash incoming addresses across up to ten gateway blocks forming a *CML Port Aggregation Group* (CPAG). RN SAM can support up to 16 CPAGs (RNSAM_NUM_CPA_GRP <= 16). RN SAM also has an address mask for each CPA which can be used to remove certain bits from the hashing.

For example, to stripe the incoming address at 512B granularity, the address mask bits[51:6] can be set to 0xFFFFFFFFFFF8. With this mask setup, the logical operator AND is applied to all incoming addresses before hashing the address bits. For information about programming, see the RN SAM registers.

RNSAM supports maximum of 32 CXG or CCG target IDs. Each CPAG derives the PAG target ID based on the address hashing and the hash index is looked up into the CXG/CCG target ID table. Base index for each CPAG is derived based on the linked list mechanism (RNSAM_FLEX_TGTID_EN = 1).

Legacy CMN Mode (CPAG = 5, pag_tgtid = 10)

For Legacy CMN mode, need to reset the parameter RNSAM_FLEX_TGTID_EN = 0.

The following table shows the number of CXG/CCG targets that are allowed in each CPAG.

Table 2-37 Number of CXG/CCG targets allowed in each CPAG

CPA Group ID	Number of CXGs/CCGs supported
CPAG_0	8
CPAG_1	2
CPAG_2	4
CPAG_3	2
CPAG_4	2

CPAGs use a shared target ID table with a maximum total of ten target IDs. Therefore the number of CPAGs that are available depends on the number targets that are assigned to each CPAG. For example, if CPAG 0 is hashing across eight ports, then CPAG 1, CPAG 2, and CPAG 3 must not be used. However, CPAG 4 can contain the two remaining target IDs.

The following table shows the mapping of target IDs to each CPAG.

Table 2-38 Mapping of target IDs to CPAGs

cml_port_aggr_grp_reg fields	CPAG_0	CPAG_1	CPAG_2	CPAG_3	CPAG_4
pag_tgtid0	Y	-	-	-	-
pag_tgtid1	Y	-	-	-	-
pag_tgtid2	Y	Y	-	-	-
pag_tgtid3	Y	Y	-	-	-
pag_tgtid4	Y	-	Y	-	-
pag_tgtid5	Y	-	Y	-	-
pag_tgtid6	Y	-	Y	Y	-
pag_tgtid7	Y	-	Y	Y	-
pag_tgtid8	-	-	-	-	Y
pag_tgtid9	-	-	-	-	Y

Support for CPA in HN-F SAM

HN-F also supports CPA for snoop requests going to a remote chip. HN-F uses the same hashing as RN SAM so that a given address always goes to the same CML gateway block.

HN-F uses the *Logical ID* (LDID) of the RN-F to determine whether CPA is enabled. As [2.4.14 Cross chip routing and ID mapping on page 2-155](#) describes, HN-F contains the LDID to physical node ID conversion table as [Table 2-54 Example program on page 2-159](#) shows in the Example Programming table. This table, along with the CHI node ID and valid fields, also contains remote, cpa_en, and cpa_grpid bits. HN-F uses these bits to determine whether the RN-F is enabled to use CPA. It then sends the snoops through appropriate ports by hashing the address bits. To enable CPA for the ID of each RN-F, see the logical to physical ID conversion registers in HN-F.

Guidelines for enabling CPA in RN SAM and HN-F SAM

When enabling CPA in RN SAM and HN-F SAM, the following rules apply:

- CPA can only be used for SCG and non-hashed address ranges in the RN SAM
- CPA must not be enabled for Device traffic. For more information, see the *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*.
- Each non-hashed memory range can be explicitly enabled to use CPA or use a single non-hashed target ID

- The target ID of each SCG can be explicitly enabled to use CPA
- The HN-F, when participating as a CPA target for traffic from a remote RN-F, must not receive non-CPA traffic from the same remote RN-F. This requirement means that an RN-F cannot send CPA and non-CPA traffic to the same remote HN-F.
- Each SCG in RN SAM can contain only one CPA group along with local HN-F target IDs
- The `cml_port_aggr_grp_reg0` and `cml_port_aggr_grp_reg1` registers contain the full list of CXG target IDs

For the High BW streaming PCIE writes to the remote memory over CCIX, CPAG hashing is disabled and the HTG is used to directly configure the AXID based hashing and programing the CCG nodes to HNP target ID table.

AXID based hashing is not supported inside of the CPAG.

Address bit masking in the RN SAM

The CMN-700 RN SAM supports masking of address bits used for range compare and address hashing.

Address bit masking in the RN SAM can be enabled by programming the following registers:

- `rnsam_hash_addr_mask_reg`
- `rnsam_region_cmp_addr_mask_reg`

For more information about these registers, see [rnsam_hash_addr_mask_reg on page 3-699](#) and [rnsam_region_cmp_addr_mask_reg on page 3-701](#).

When RN SAM compares the incoming address against the programmed ranges, it uses different address bit ranges for different region types:

- For hashed and non-hashed memory regions, RN SAM uses address bits [MSB:26].
- For GIC memory region, RN SAM uses address bits [MSB:16].

By programming select bits to `0b0` in the `rnsam_region_cmp_addr_mask_reg` mask register, both the incoming address and the programmed address ranges can be masked off before comparison. This region mask is applied to hashed, non-hashed, and GIC memory regions.

RN SAM hashes all address bits [MSB:6] to equally distribute the requests across all HN-F and SN-F target devices. By programming select bits in the `rnsam_hash_addr_mask_reg` address mask register to `0b0`, those address bits can be removed from the hashing logic. This feature is only applicable to hashed memory regions.

The following limitations apply:

- If 3-SN, 5-SN, or 6-SN mode is enabled, address bits [16:8] and the `top_addr_bits` are essential in distributing the addresses between memory. Arm recommends that these bits are masked carefully to avoid memory aliasing.
- Range compare mask must not mask off bits that represent the size of the region. For example, if any of the region sizes are 64MB, address bit 26 cannot be masked. Similarly, if a region size is 512MB, address bit 29 cannot be masked.
- The address bits masked in HN-F and RNSAM must be consistent. This requirement ensures that PrefetchTgt requests from an RN-F to SN-F are addresses by the same SN-F as SLC miss from the HN-F to SN-F.

For hased and non-hased memory regions, RN SAM uses address bits [MSB:16].

2.4.6 CXRA/CCRA SAM

All *Requesting Agents* (CXRA/CCRA) in CMN-700 require a *Requesting Agent System Address Map* (RA SAM) to determine the target *Home Agent ID* (HAID). This HAID is used as the target ID to route the Request.

The RA SAM uses configuration registers to specify address regions and corresponding HAIDs. Each address region is configured by programming the base address and corresponding size of the address

region (or programming the address limit). Each valid address region is marked using a valid bit. The incoming address is compared against programmed valid address regions to generate a specific HAID.

The following figure shows a RA SAM block diagram.

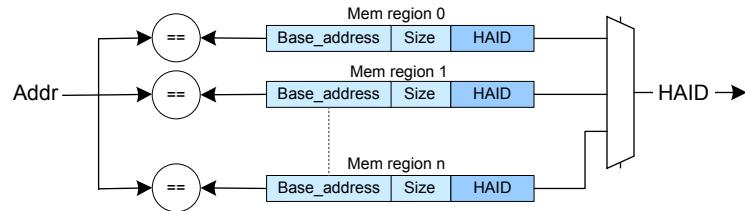


Figure 2-52 RA SAM block diagram

Address region requirements

Each of the programmed address region sizes must be a power of two and must be naturally-aligned to its size. For example, a 1GB partition must start at 1GB boundary. That is, 0GB-1GB or 1GB-2GB and so forth, but it cannot start from 1.5GB or 2.5GB.

For details on how CCIX messages are routed based on the CXRA/CCRA SAM programming, see [2.4.14 Cross chip routing and ID mapping on page 2-155](#).

2.4.7 HN-F SAM

Transactions from an HN-F to an SN must pass through an HN-F SAM to generate a CHI target ID.

The HN-F SAM programming sequence is described in [3.4.3 RN SAM and HN-F SAM programming on page 3-1239](#).

The following figure shows how the target ID is determined from the HN-F SAM.

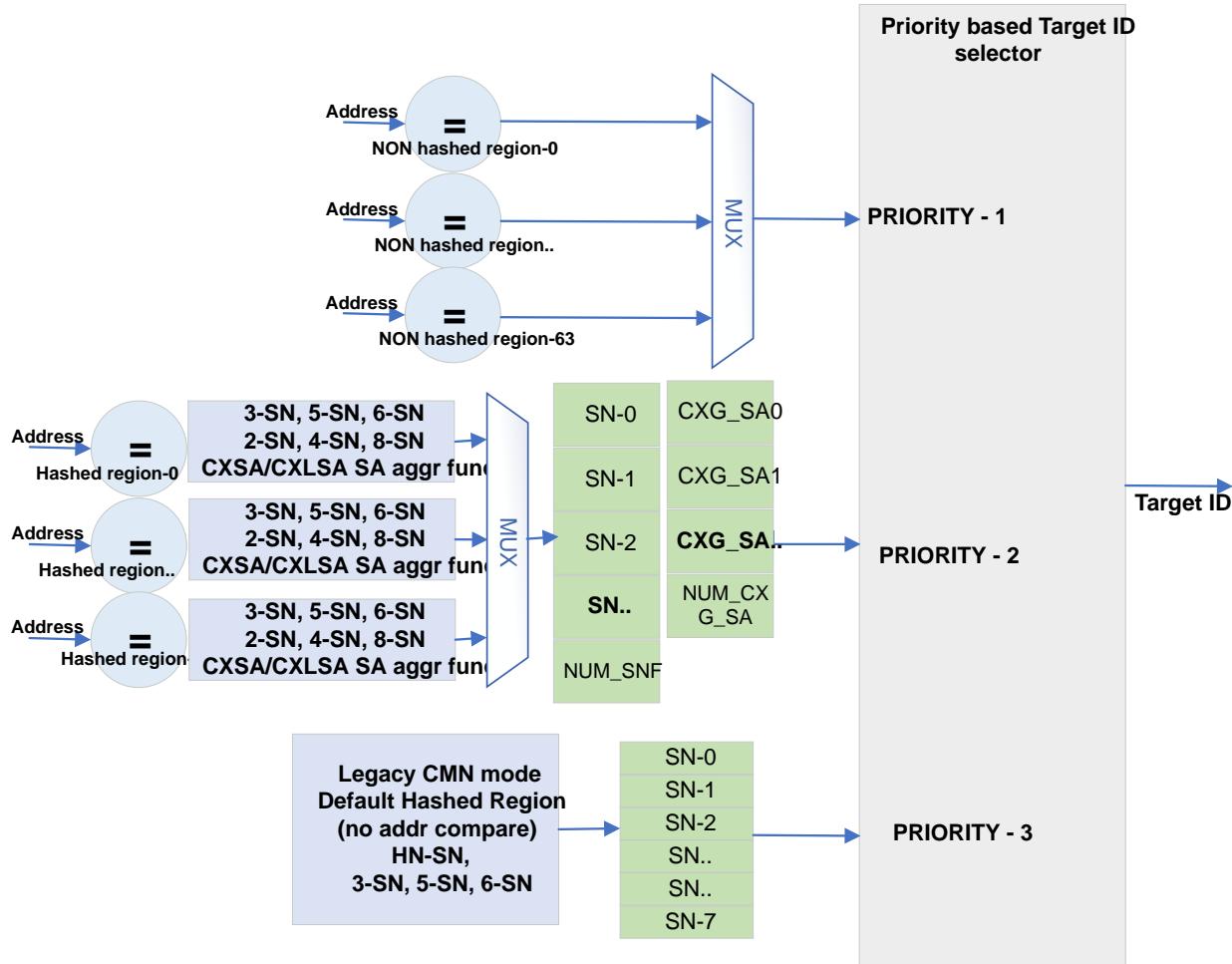


Figure 2-53 HN-F SAM target ID selection policy

The preceding figure shows that the HN-F SAM has the following mapping policies to generate target IDs for transactions:

- Range-based:
 - Non-hashed
 - Hashed target groups
- Default hashed region (Legacy CMN mode):
 - Hashed
 - Direct mapping

An HN-F cannot use both hashed mapping mode and direct mapping mode. To map transactions that fall in the range-based part of the HN-F SAM, HN-F SAMs support priority-based target ID selection. The order of priority when selecting a target ID is:

1. Range-based: non-hashed SN target ID
2. Range-based: hashed SN target ID
3. Default Hashed region: Hashed target ID or direct mapped target ID (Legacy CMN mode).

Range-based mapping: Non-hashed target IDs

Non-hashed range-based mapping is an address-based unique target ID generation policy. Up to 64 memory regions (`HNSAM_NUM_NONHASH <= 64`) can be created, each targeting a single SN. This mode is useful where a partition of memory from the global DRAM is mapped explicitly to an individual SN, for example, an on-chip SRAM.

Range-based mapping: Hashed target groups

Hashed range-based mapping is an address based hashed target ID generation policy. Up to eight memory regions can be created, each targeting a group of SN's. This mode is useful where a partition of memory from the global DRAM is mapped explicitly to a group of SN. These modes can only be used when a DRAM partition targets the following SN configurations:

- 3-SN mode Addresses from a hashed region are striped across three SNs
- 5-SN mode Addresses from a hashed region are striped across five SNs
- 6-SN mode Addresses from a hashed region are striped across six SNs
- 2-SN mode Addresses from a hashed region are striped across two SNs
- 4-SN mode Addresses from a hashed region are striped across four SNs
- 8-SN mode Addresses from a hashed region are striped across eight SNs

Addresses are striped at a 256B granularity between the 3SNs/5SNs/6SNs. The stripe function uses address bits [16:8], and an extra two (3-SN) or three (5-SN, 6-SN) user defined address bits.

Addresses are striped at a 64B granularity between the 2SNs/4SNs/8SNs. The stripe function uses XOR of address bits [MSB:6].

- Two SNs:
 - Number of bits in select: 1
 - select [0] = $(6^7^8^{\dots}51)$
- Four SNs:
 - Number of bits in select: 2
 - select [0] = $(6^8^10^{\dots}50)$
 - select [1] = $(7^9^11^{\dots}51)$
- Eight SNs:
 - Number of bits in select: 3
 - select [0] = $(6^9^12^{\dots}51)$
 - select [1] = $(7^10^13^{\dots}49)$
 - select [2] = $(8^11^14^{\dots}50)$

Default Hashed region: Hashed mapping (legacy CMN mode)

The HN-F SAM supports the following hashed modes for SN target ID selection. The default hashed region don't have any address comparison. when incoming address do not match range based mapping, then SN TgtID is derived from the default hashed region.:.

- | | |
|------------------|---|
| 3-SN mode | Addresses from a given HN-F are striped across three SNs. |
| 5-SN mode | Addresses from a given HN-F are striped across five SNs. |
| 6-SN mode | Addresses from a given HN-F are striped across six SNs. |

Direct mapping

Direct SN mapping is used if the SCG targets 1, 2, 4, 8, 16, or 32 SNs. In this case, the transaction uses the SN0 target ID. Distributing accesses across the SNs targeted by the SCG is achieved by programming the SN0 field of each HN-F to the SN node ID it targets. For example, if an SCG with eight HN-Fs targets eight SNs, the SN0 field of each HN-F would be programmed with a different SN node ID. If that same SCG with eight HN-Fs targets four SNs instead, every two HN-F nodes would have the same SN0 field value.

Aggregated SA selection function

HNSAM supports hashing across multiple CXL.MEM & CCIX CXSA devices. These devices are selected based on the aggregated SA selection function defined in CCIX 1.2_ECR_80_SA_Aggregation.

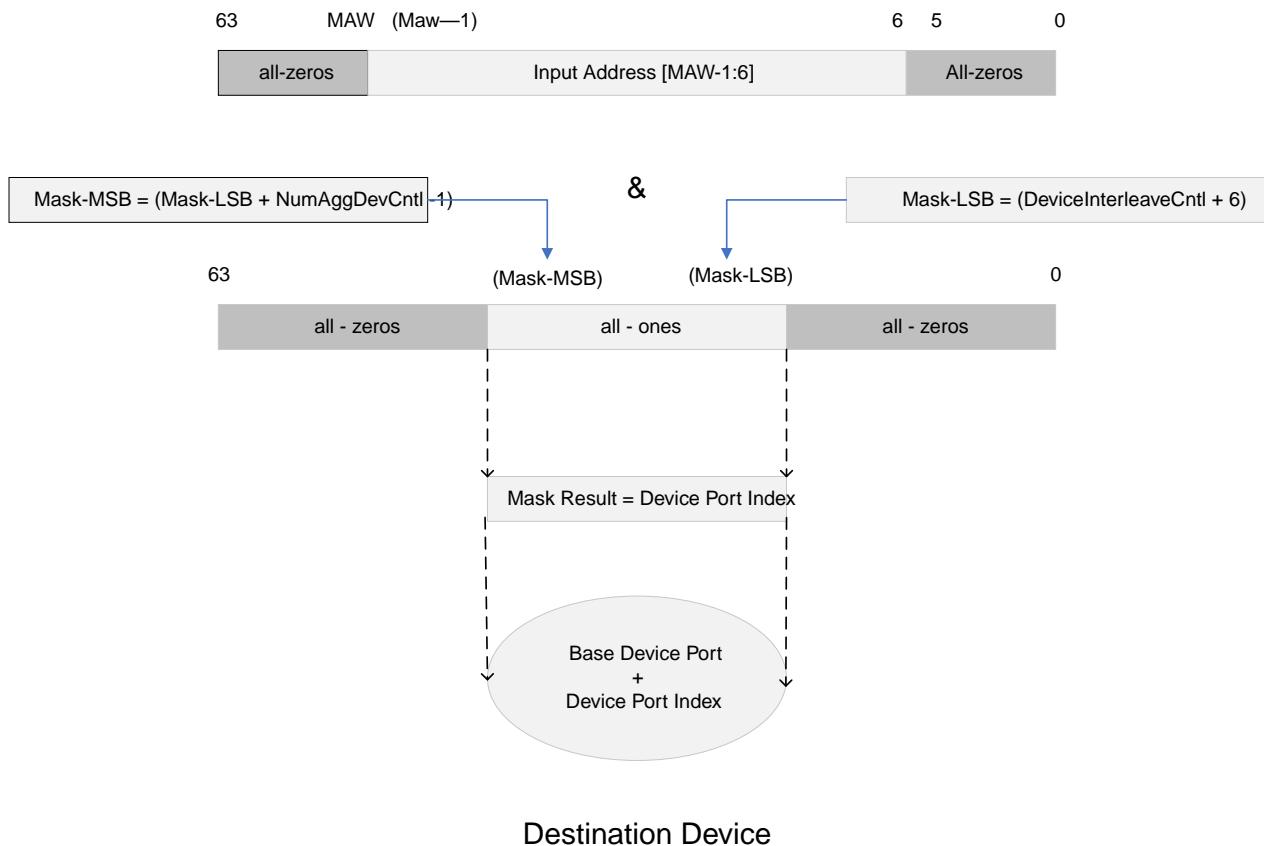


Figure 2-54 Aggregated SA selection function

Aggregated SA selection needs the following configurations to determine the SN targetID:

- Device interleaving: This configuration controls the interleave size across all aggregated CXSA/CXLSA Devices, supports interleave sizes between 64 Bytes to 2 Mbytes. This controls the number of LSB bits to be stripped from the address.
- Aggregated devices count: This configuration controls the number of the CXSA/CXLSA devices aggregated, supports aggregated devices are 1,2,4,8,16. This controls the number of MSB bits to be stripped from the address.

Address bits extracted by stripping the LSB bits and MSB bits based on the above configuration yields the selection bits for deriving CXSA/CXLSA target ID.

HNSAM range-based regions support two ways of defining a memory region, it is selected through a user parameter (HNSAM_RCOMP[MW1][SKM2]_EN).

- Base address & region size, size of the region being power of two (64MB minimum region size).
- Lower address & upper address, no restrictions on size of the region. (RCOMP_EN = 1)

Minimum size of the memory region is selected through a user parameter that defines the lower address bit for the comparison. This is limited to the lower address & upper address mode only.

PARAM: HNSAM_RCOMP_LSB – allowed values [20-26]

- HNSAM_RCOMP_LSB = 20, defines minimum memory size = 1 MB
- HNSAM_RCOMP_LSB = 21, defines minimum memory size = 2 MB

- HNSAM_RCOMP_LSB = 22, defines minimum memory size = 4 MB
- HNSAM_RCOMP_LSB = 26, defines minimum memory size = 64 MB

SN Target-ID tables

HNSAM hashed target groups use the Target-IDs from the Target-ID structure with depth determined by number of SNs in the configuration. Maximum SN target ID entries supported are 64.

HNSAM hashed target groups use CXSA/CXLSA Target IDs from the separate Target-ID structure with depth determined by number of SNs in the configuration. Maximum CCG SA target ID entries supported are 16.

Each hashed target group generates the index based on the address striping and lookup into the target ID tables for the SN/CCG target ID. Base index for each HTG is derived based on the linked list mechanism from the boot time programming.

Hashed region[n] base index = Hashed region [n-1] base index + Hashed region[n-1] num SNF

HN-F to SN-F memory striping in HN-F SAM

The CMN-700 HN-F SAM supports three memory striping modes, which are known as 3-SN mode, 5-SN mode, and 6-SN mode respectively. In these modes, the HN-F stripes addresses across across three SN-Fs, five SN-Fs, or six SN-Fs respectively.

3-SN mode

In 3-SN mode, a stripe function ensures that traffic is distributed evenly among the three SNs. The stripe function is based on PA[16:8] and two higher bits in the PA. The two higher PA bits are referred to as top_address_bit1 and top_address_bit0. The top address bits are selected so that three of the four combinations of the top address bits appear evenly in the selected address space, and the fourth combination never appears.

————— Note —————

In some situations, a top bit can be the inverse of the selected PA bit.

For each physical address, one of the three SNs is selected using the following formula:

$$\text{SN} = \{ \text{ADDR}[10:8] + \text{ADDR}[13:11] + \text{ADDR}[16:14] + ((\text{top_addr_bit1} << 1) | \text{top_addr_bit0}) \} \% 3$$

General SN distribution behavior example

For a simple case with a 3GB flat address space starting at address 0x0, top_address_bit1 is PA[31], and top_address_bit0 is PA[30]. With increasing physical address, the function steps between SNs at a 256-byte granularity. As the physical address iterates from 0-128KB, with top_address_bit1 = top_address_bit0 = 0, the first three terms distribute the traffic relatively evenly among the SNs. Of the 512 blocks (256B each) in the first 128KB, the distribution is:

SN[0]	170 blocks 33.2%
SN[1]	171 blocks 33.4%
SN[2]	171 blocks 33.4%

This pattern repeats over each 128KB until 1GB, where top_address_bit0 toggles. With top_address_bit1 = 0 and top_address_bit0 = 1, the pattern is shifted. For each 128KB:

SN[0]	171 blocks 33.4%
SN[1]	170 blocks 33.2%
SN[2]	171 blocks 33.4%

At 2GB, when top_address_bit1 = 1 and top_address_bit0 = 0, the pattern shifts again:

SN[0]	171 blocks 33.4%
--------------	------------------

SN[1]	171 blocks 33.4%
SN[2]	170 blocks 33.2%

Over the full 3GB, the same number of lines are distributed to each SN.

The HN-F uses the hn_cfg_three_sn_en bit in its por_hnf_sam_control register to enable routing to three SNs. In the por_hnf_sam_control register, the hn_cfg_sam_top_address_bit0 and hn_cfg_sam_top_address_bit1 fields must be configured at boot time. These two address bits are decoded, and used with a hashing function to determine the target SN-F.

5-SN and 6-SN modes

Similar to 3-SN hashing, the 5-SN and 6-SN modes extend the function to equally distribute addresses between five or six SNs respectively. For each physical address, one of the SNs is selected using the following formula:

$$SN = \{ ADDR[10:8] + ADDR[13:11] + ADDR[16:14] + ((top_addr_bit2<<2) | (top_addr_bit1<<1) | top_addr_bit0) \} \% 6$$

HN-F SAM uses the following bits in the por_hnf_sam_control register to enable striping across five or six SN-Fs:

- hn_cfg_five_sn_en
- hn_cfg_six_sn_en

In 5-SN and 6-SN mode, HN-F SAM also uses hn_cfg_sam_top_address_bit2 field in the por_hnf_sam_control register along with hn_cfg_sam_top_address_bit1 and hn_cfg_sam_top_address_bit0 to hash the incoming address.

3-SN, 5-SN, and 6-SN configurations

The following table shows the valid top address bits for Arm PDD Memory Map. For more information, refer to the *Principles of Arm® Memory Maps White Paper*. This configuration ensures equal distribution of requests across all SN-Fs and prevents memory aliasing. The SAM also provides an inv_top_address_bit configuration bit, which can be used with top address bits as the following table shows.

Table 2-39 3-SN mode top address bits[bit 1, bit 0]

Combination 1 (inv_top_address_bit set to 0b0)	Combination 2 (inv_top_address_bit set to 0b0)	Combination 3 (inv_top_address_bit set to 0b1)	Combination 4 (inv_top_address_bit set to 0b1)
[0, 0]	[1, 1]	[0, 0]	[0, 0]
[0, 1]	[0, 1]	[1, 0]	[0, 1]
[1, 0]	[1, 0]	[1, 1]	[1, 1]

Note

When inv_top_address_bit=1, it forces the SAM to invert the top most significant top address bit. For 3-SN mode, top_address_bit1 is inverted. For 5-SN and 6-SN mode, top_address_bit2 is inverted.

The following table shows the valid combinations for the address bits for 6-SN mode with PDD memory map.

Table 2-40 6-SN mode top address bits[bit 2, bit 1, bit 0]

Combination 1 (inv_top_address_bit set to 0b0)	Combination 2 (inv_top_address_bit set to 0b0)	Combination 3 (inv_top_address_bit set to 0b1)
[0, 0, 0]	[0, 1, 0]	[0, 0, 0]
[0, 0, 1]	[0, 1, 1]	[0, 0, 1]
[0, 1, 0]	[1, 0, 0]	[0, 1, 0]
[0, 1, 1]	[1, 0, 1]	[0, 1, 1]
[1, 0, 0]	[1, 1, 0]	[1, 1, 0]
[1, 0, 1]	[1, 1, 1]	[1, 1, 1]

The combinations for 5-SN mode must follow similar rules to 6-SN mode programming. The top address bit combinations must be five sequential combinations from the preceding table.

Example 2-8 Example for PDD memory map

Assume that a system supports three SN-Fs with 32GB of DRAM at each SN-F port and all three SN-Fs are used for SCG 0. Since the DRAM space is non-contiguous in this memory map (2GB + 30GB + 480GB), the base addresses for each DRAM partition are:

1. a. 000_8000_0000 to 000_FFFF_FFFF (2GB)
b. 008_8000_0000 to 00F_FFFF_FFFF (30GB)
2. 088_0000_0000 to 08F_FFFF_FFFF (32GB)
3. 090_0000_0000 to 097_FFFF_FFFF (32GB)

The first two regions together comprise a 32GB region, while the remaining two regions are 32GB each. The following table breaks down the address bits for the regions that are shown in the preceding list.

Table 2-41 Address region example bit settings: 3-SN example

Region	39	38	37	36	35	34	33	32	31
1	0	0	0	0	0	0	0	0	1
1	0	0	0	0	1	x	x	x	1
2	1	0	0	0	1	x	x	x	x
3	1	0	0	1	0	x	x	x	x

From the address bit breakdown, the selected top address bits must make sure both regions that are marked as Region 1 have the same values. This requirement ensures no aliasing in memory. For this memory map and DRAM size, there are no address bits that directly give Combination 1 or Combination 2 as shown in [Table 2-39 3-SN mode top address bits\[bit 1, bit 0\] on page 2-130](#). However, if bits[39, 36] are used along with inv_top_address_bit = 1, then Combination 3 is possible. This approach ensures that the memory requests are equally distributed across the three SN-Fs without memory aliasing.

The following figure shows the Arm proposed memory map.

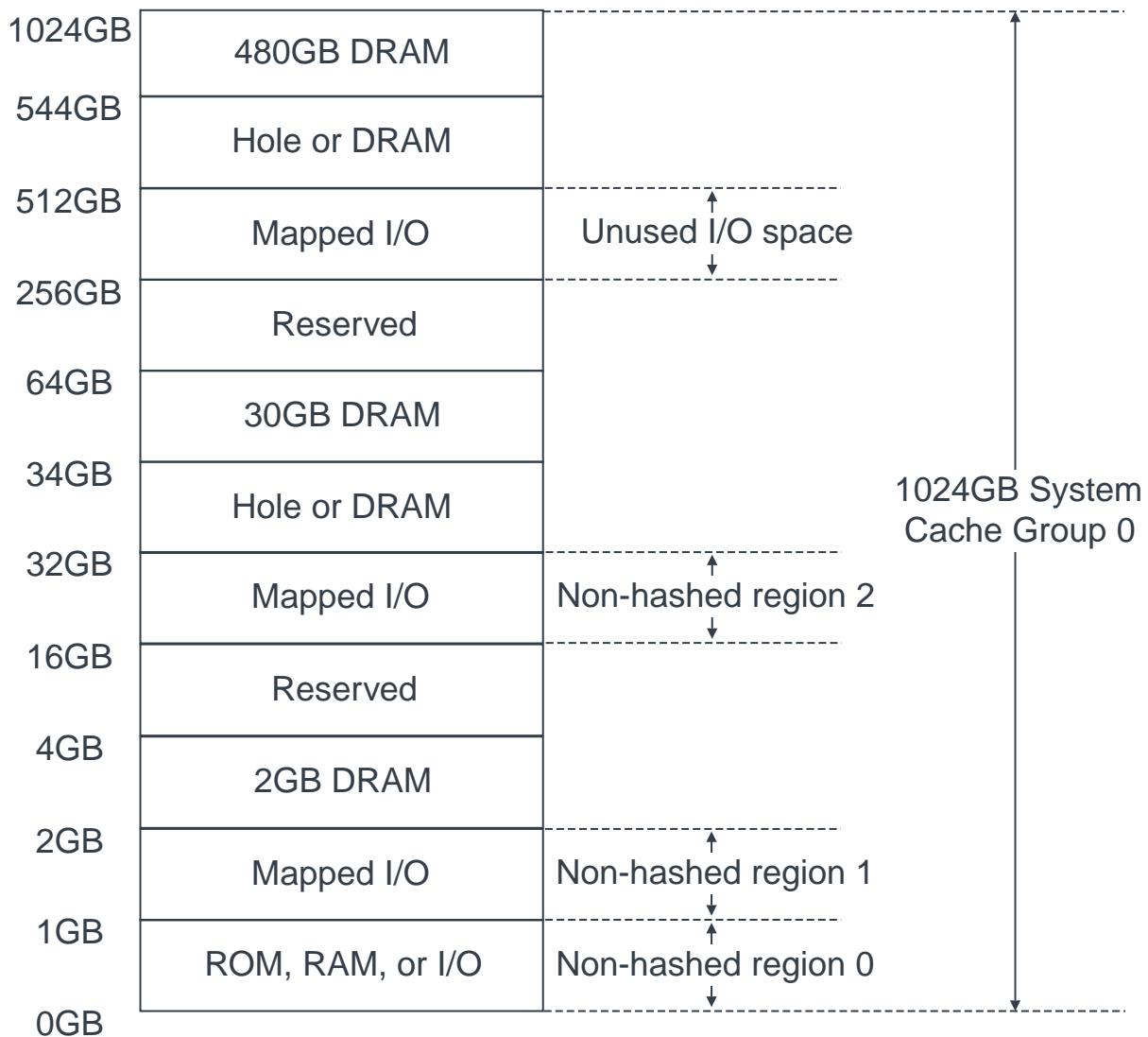


Figure 2-55 Example memory map programming

The following tables provide example address bits that provide equal distribution of memory across all SN-Fs in 3-SN and 6-SN hashed modes.

Table 2-42 3-SN DRAM size settings

3-SN DRAM size at each SN-F port	Top address bits[bit 1, bit 0]	Inv_top_address_bit value
1GB (Total 3GB)	[35, 30]	0b0
2GB (Total 6GB)	[35, 31]	0b0
4GB (Total 12GB)	[33, 32]	0b0
8GB (Total 24GB)	[34, 33]	0b0
16GB (Total 48GB)	[39, 34]	0b0
32GB (Total 96GB)	[39, 36]	0b1

Table 2-42 3-SN DRAM size settings (continued)

3-SN DRAM size at each SN-F port	Top address bits[bit 1, bit 0]	Inv_top_address_bit value
64GB (Total 192GB)	[37, 36]	0b0
128GB (Total 384GB)	[38, 37]	0b0

Table 2-43 6-SN DRAM size settings

6-SN DRAM size at each SN-F port	Top address bits[bit 2, bit 1, bit 0]	Inv_top_address_bit value
1GB (Total 6GB)	[35, 31, 28]	0b0
2GB (Total 12GB)	[33, 32, 28]	0b0
4GB (Total 24GB)	[34, 33, 28]	0b0
8GB (Total 48GB)	[39, 34, 33]	0b0
16GB (Total 96GB)	[39, 36, 28]	0b1
32GB (Total 192GB)	[37, 36, 28]	0b0
64GB (Total 384GB)	[38, 37, 28]	0b0

The top address bit combinations for 5-SN mode are not available with the PDD memory map for all DDR size combinations. For specific memory maps, you must follow the 6-SN rules to achieve valid address bit combinations and contiguous SN addresses.

SN contiguous address spaces

This section describes which physical address bits must be connected to an SN-F for various configurations.

If all HN-Fs send their cache misses to a single SN-F, that SN-F sees the full address space. However, it is common for a system to have two or more SN-Fs, with each HN-F sending its cache misses to a single SN-F. In this scenario, each SN-F receives only part of the address space. SN-F typically removes one or more address bits to retain a contiguous address map. The full physical address is presented to each SN-F for every request so that any SN-F based memory protection logic can function. However, the actual mapping to RAM locations can be done with the modified address. The address modification depends on multiple factors:

- Number of HN-Fs in the cache group
- Number of SN-Fs in the cache group
- Which HN-Fs share SN-Fs

2ⁿ-SN address striping

The following table provides HN-F and SN-F combinations that are supported within a cache group, along with the address bits that should be removed.

Table 2-44 HN-F and SN-F combinations supported within a cache group

Number of HN-Fs	Number of SN-Fs	Bits to strip from full PA
2	1	None
	2	[6]
4	1	None
	2	[7]
	4	[7, 6]

Table 2-44 HN-F and SN-F combinations supported within a cache group (continued)

Number of HN-Fs	Number of SN-Fs	Bits to strip from full PA
8	1	None
	2	[8]
	4	[8, 7]
	8	[8, 7, 6]
16	1	None
	2	[9]
	4	[9, 8]
	8	[9, 8, 7]
	16	[9, 8, 7, 6]
32	1	None
	2	[10]
	4	[10, 9]
	8	[10, 9, 8]
	16	[10, 9, 8, 7]
	32	[10, 9, 8, 7, 6]
64	1	None
	2	[11]
	4	[11, 10]
	8	[11, 10, 9]
	16	[11, 10, 9, 8]
	32	[11, 10, 9, 8, 7]

The method that is used to calculate the bits stripped is as follows:

1. The highest bit removed is the least significant address bit used in the XOR function for the most significant bit of the HN-F select hash function.

64 HN-Fs PA[11]

32 HN-Fs PA[10]

16 HN-Fs PA[9]

Eight HN-Fs PA[8]

Four HN-Fs PA[7]

Two HN-Fs PA[6]

2. The number of bits stripped is $\log_2(\text{number of SN-Fs})$, sequentially below the highest bit.

This approach to bit stripping assumes that HN-Fs that share SN-Fs are sequential in the RN SAM cache group HN-F table. For example, if there are eight HN-Fs and two SN-Fs, the bottom four HN-Fs in the RN SAM table would share an SN-F. The top four HN-Fs would share an SN-F.

3-SN and 6-SN address striping

As 3-SN and 6-SN address hashing implements modulo function according to the top address bits used, the SN-F must remove these bits to achieve a contiguous memory map in the DRAM.

3-SN mode The SN-F must remove top_address_bit1 and top_address_bit0.

6-SN mode The SN-F must remove top_address_bit2, top_address_bit1, and top_address_bit0.

3-SN, 5-SN, and 6-SN address striping

As 3-SN, 5-SN, and 6-SN address hashing implements modulo function according to the top address bits used, the SN-F must remove these bits to achieve a contiguous memory map in the DRAM.

3-SN mode The SN-F must remove top_address_bit1 and top_address_bit0.

5-SN mode The SN-F must remove top_address_bit2, top_address_bit1, and top_address_bit0.

6-SN mode The SN-F must remove top_address_bit2, top_address_bit1, and top_address_bit0.

Address bit masking in the HN-F SAM

CMN-700 supports masking of address bits in the HN-F SAM. Certain restrictions apply to this process.

HN-F SAM

HN-F SAM uses address bits [MSB:26] when comparing the incoming address against the programmed ranges. By programming select bits to `0b0` in the `hnf_sam_region_cmp_addr_mask_reg` mask register, the incoming address and the programmed address ranges can be masked off before comparison. This region mask is only applicable to region-based memory partitioning in the HN-F and so the mask is not applied to the hashing scheme in 3-SN, 5-SN, and 6-SN modes.

Address bit masking

HN-F SAM supports masking of address bits used for 3-SN, 5-SN, or 6-SN address hashing. This feature can be enabled by programming the `hn_sam_hash_addr_mask_reg`.

HN-F SAM

HN-F SAM uses address bits [MSB:26] when comparing the incoming address against the programmed ranges. By programming select bits to `0b0` in the `hnf_sam_region_cmp_addr_mask_reg` mask register, the incoming address and the programmed address ranges can be masked off before comparison. This region mask is only applicable to region-based memory partitioning in the HN-F and so the mask is not applied to the hashing scheme in 3-SN and 6-SN modes.

Address bit masking

HN-F SAM supports masking of address bits used for 3-SN or 6-SN address hashing. This feature can be enabled by programming the `hn_sam_hash_addr_mask_reg`.

The following limitations apply:

- Range compare mask must not mask off bits that represent the size of the region. For example, if any of the region sizes are 64MB, address bit 26 cannot be masked. Similarly, if a region size is 512MB, address bit 29 cannot be masked.
- If 3-SN, 5-SN, or 6-SN mode is enabled, address bits [16:7] and the `top_addr_bits` are essential in distributing the addresses between memory. We recommend that these bits are masked carefully to avoid memory aliasing.
- The address bits masked in HN-F and RN SAM must be consistent. This requirement ensures that PrefetchTgt requests from an RN-F to SN-F are addresses by the same SN-F as SLC miss from the HN-F to SN-F.

User defined hashing logic

The hash modules can be selectively replaced to implement user defined hash logic. HN SAM implements user defined configuration registers to be used for the user defined hash logic. The number of user defined registers are enabled using a user parameter (`HNSAM_CUSTOM_REGS` [0-2]).

Refer to the CIM for more information on modifying the hash modules.

2.4.8 SAM memory region size configuration

Hashed, non-hashed, and GIC memory regions support various sizes. Each memory partition must be individually programmed in the SAM registers.

RN SAM and HN-F SAM support the following memory partition sizes:

Hashed and non-hashed	64MB to maximum addressable space (2^{48}). 64MB for RN SAM and HN-F SAM up to maximum addressable space ($2^{\text{PA_WIDTH}}$).
GIC	64KB, 128KB, 256KB, and 512KB.

The following table lists the memory partition size encodings that are used to program the RN SAM and HN-F SAM registers.

Table 2-45 RN SAM and HN-F SAM configuration register memory partition sizes

Memory partition size		regionX_size value
GIC	Hashed and non-hashed	
64KB	64MB	3'b000 (GIC) / 7'b0000000 (other)
128KB	128MB	3'b001 (GIC) / 7'b0000001 (other)
256KB	256MB	3'b010 (GIC) / 7'b0000010 (other)
512KB	512MB	3'b011 (GIC) / 7'b0000011 (other)
N/A	1GB	7'b0000100
	2GB	7'b0000101
	4GB	7'b0000110
	8GB	7'b0000111
	16GB	7'b0001000
	32GB	7'b0001001
	64GB	7'b0001010
	128GB	7'b0001011
	256GB	7'b0001100
	512GB	7'b0001101
	1TB	7'b0001110
	2TB	7'b0001111
	4TB	7'b0010000
	8TB	7'b0010001
	16TB	7'b0010010
	32TB	7'b0010011
	64TB	7'b0010100
	128TB	7'b0010101
	256TB	7'b0010110
	512TB	7'b0010111
	1PB	7'b0011000
	2PB	7'b0011001
	4PB	7'b0011010

The RN SAM also outputs the target type of the device along with the target ID for the RN to use in various optimizations. The target type encodings are listed in the following table.

Table 2-46 Device target types

Device type	Target type
HN-F	2'b00
HN-I	2'b01
CXRA	2'b10
Reserved	2'b11

The following table contains RA SAM configuration register memory partition sizes and encodings.

Table 2-47 RA SAM configuration register memory partition sizes

Memory partition size	regionX_size value
64KB	6'b000000
128KB	6'b000001
256KB	6'b000010
512KB	6'b000011
1MB	6'b000100
2MB	6'b000101
4MB	6'b000110
8MB	6'b000111
16MB	6'b001000
32MB	6'b001001
64MB	6'b001010
128MB	6'b001011
256MB	6'b001100
512MB	6'b001101
1GB	6'b001110
2GB	6'b001111
4GB	6'b010000
8GB	6'b010001
16GB	6'b010010
32GB	6'b010011
64GB	6'b010100
128GB	6'b010101
256GB	6'b010110
512GB	6'b010111
1TB	6'b011000
2TB	6'b011001
4TB	6'b011010

Table 2-47 RA SAM configuration register memory partition sizes (continued)

Memory partition size	regionX_size value
8TB	6'b011011
16TB	6'b011100
32TB	6'b011101
64TB	6'b011110
128TB	6'b011111
256TB	6'b100000
512TB	6'b100001
1PB	6'b100010
2PB	6'b100011
4PB	6'b100100

2.4.9 HN-I SAM

To simplify mapping and ordering of downstream endpoint address space, the HN-I SAM maps an incoming address to a target endpoint that is connected downstream behind HN-I.

The endpoint can be one of the following:

- Peripheral with memory-mapped I/O space, such as UART or GPIO
- Physical memory, such as SRAM or FLASH

———— **Restriction** ———

Arm recommends that the HN-I SAM is only programmed during the boot process.

To map and order the address space of these endpoints, the HN-I SAM supports:

- Up to three Address Regions

———— **Restriction** ———

Address Regions 1, 2, and 3 must not overlap.

- One Order Region of configurable size for each Address Region
- A default Address Region, Address Region 0

Each Address Region can be programmed as either peripheral or physical memory.

———— **Note** ———

By default, each Address Region is mapped to peripheral memory.

Physical

- Follows normal memory ordering guarantees.
- Order Region programming function output does not matter.

Peripheral

- Follows device memory ordering guarantees.
- These Address Regions can be further divided into smaller address spaces that are known as Order Regions. Device memory ordering guarantees are maintained within each Order Region.
- To enforce strict ordering for a specific Address Region, program its Order Region size to 6'b111111.

Caution

If there is potential for new requests to fall into a newly configured Address Region or Order Region, and these requests require ordering with respect to the existing outstanding requests, the corresponding Address Region register must be disabled.

Note

- HN-I does not support write streaming from RN-Fs. HN-I sends CompDBID in response to RN-F write requests with ReqOrder and ExpCompAck. This response breaks the OWO as writes can be dispatched on AXI out of order.
 - OWO is still supported from RN-I, RN-D, and CXHA nodes.
-

The minimum address granularity for Address Regions and Order Regions is 4KB. This size is equivalent to the minimum slave address space granularity in AXI/ACE-Lite. Therefore, the base address in the Address Region {1, 2, 3} Configuration Registers only includes bits[REQ_ADDR_WIDTH-1:12].

Address Region 0

By default, the entire address space of a given HN-I is mapped to Address Region 0. All transactions to this region are kept in order.

The default Order Region size in Address Region 0 is 6'b111111, which covers the entire HN-I address space. The Order Region size can also be configured to:

- 6'b101000 when REQ_ADDR_WIDTH==52
- 6'b100100 when REQ_ADDR_WIDTH==48
- 6'b100000 when REQ_ADDR_WIDTH==44

Address Region 0 is always valid. Therefore, the Address Region 0 Configuration Register does not define a Valid bit.

For more details, see [por_hni_sam_addrregion0_cfg on page 3-371](#).

HN-I SAM example configuration

This example system configuration for HN-I SAM uses three Address Regions and an Order Region within each Address Region.

The following figure shows the high-level configuration of the address space and the base addresses of each Address Region.

HN-I address space	Base address
Address Region 0 (Default Region)	
Address Region 3	0x0000_0020_0000
Address Region 0 (Default Region)	0x0000_0004_0000
Address Region 2	0x0000_0002_0000
Address Region 0 (Default Region)	0x0000_0000_4000
Address Region 1	0x0000_0000_2000
Address Region 0 (Default Region)	0x0000_0000_0000

Figure 2-56 HN-I address space example

————— Note ————

In each Address Region Configuration Register, the following bitfields use the default value:

- ser_all_wr
- ser_devne_wr
- pos_early_wr_comp_en
- pos_early_rdack_en

Address Region 0

The following figure shows the example configuration for Address Region 0.

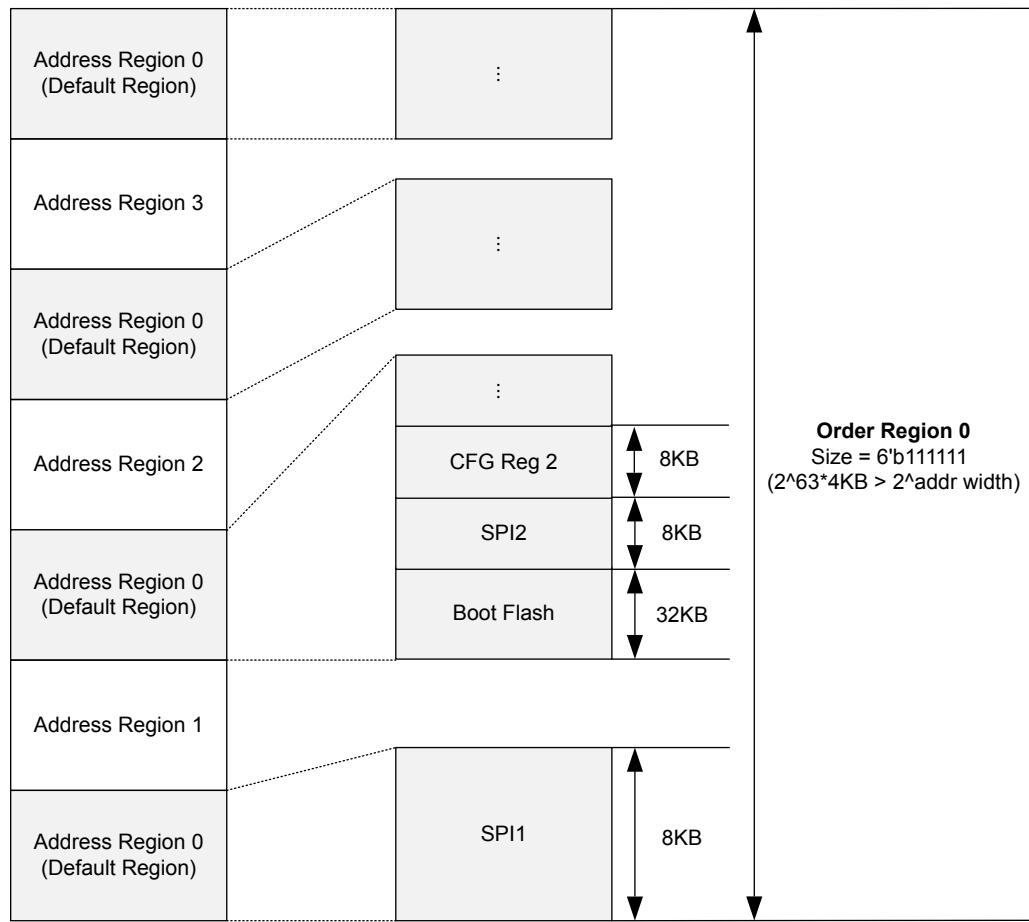


Figure 2-57 Address Region 0 configuration

The size of the maximum peripheral address space in Address Region 0 is 32KB (Boot Flash). Requests targeting this Boot Flash must be kept in order. By configuring the Order Region 0 size to 32KB, requests with addresses from `0x0000_0000_0000` to `0x0000_0000_8000` are ordered. However, since Boot Flash is not aligned to the 32KB boundary (`0x0000_0000_4000` to `0x0000_0000_C000`), requests might be out of order and cause issues. To ensure all requests to Boot Flash are ordered, the Order Region 0 size must be configured to at least 64KB. In this configuration, requests to SPI1, SPI2, and CFG Reg 2 are also ordered with respect to requests to Boot Flash. Instead, to optimize performance, Address Region 0 can have a SAM with Boot Flash aligned to the 32KB boundary and the Order Region 0 size can be configured to 32KB.

The following table shows the configured values for the Address Region 0 Configuration Register, `por_hni_sam_addrregion0_cfg`.

Table 2-48 Address Region 0 Configuration Register

Bits	Field name	Configured value
[5:0]	order_reg_size	6'h4
[58]	physical_mem_en	1'b0
[59]	ser_all_wr	1'b0

Table 2-48 Address Region 0 Configuration Register (continued)

Bits	Field name	Configured value
[60]	ser_devne_wr	1'b0
[61]	pos_early_rdack_en	1'b1
[62]	pos_early_wr_comp_en	1'b1

Address Region 1

The following figure shows the example configuration for Address Region 1.

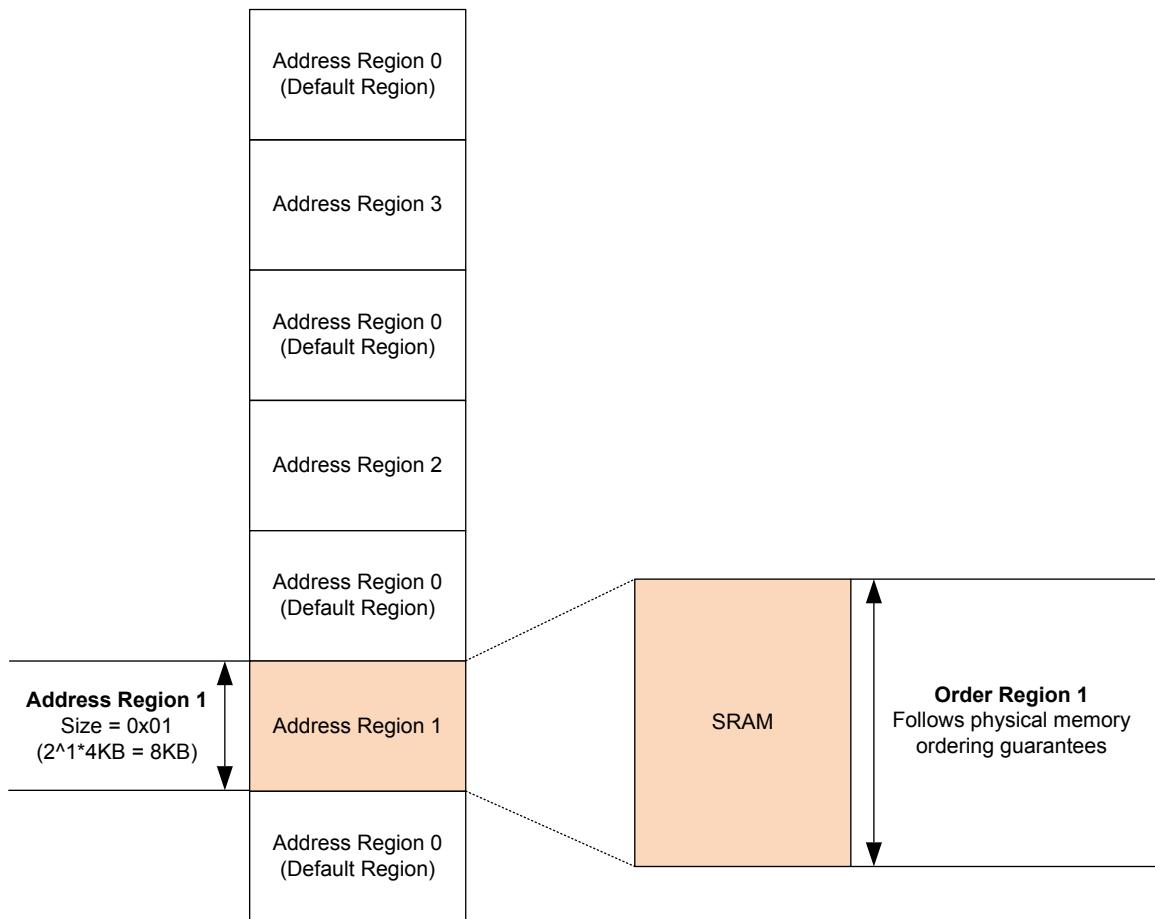


Figure 2-58 Address Region 1 configuration

Address Region 1 starts at base address `0x0000_0000_2000` and is 8KB in size. Because there is SRAM behind this region, it is mapped as physical memory. The entire Address Region 1 is considered as one Order Region. Therefore, ordering is maintained between all requests to the overlapping cache line (64B).

The following table shows the configured values for the Address Region 1 Configuration Register, `por_hni_sam_addrregion1_cfg`.

Table 2-49 Address Region 1 configuration

Bits	Field name	Configured value
[5:0]	order_reg_size	6'h1
[15:10]	addr_region_size	6'h1
[55:20]	base_addr	36'h0000_0000_2
[55:16]	base_addr	40'h0000_0000_2
[58]	physical_mem_en	1'b1
[59]	ser_all_wr	1'b0
[60]	ser_devne_wr	1'b0
[61]	pos_early_rdack_en	1'b1
[62]	pos_early_wr_comp_en	1'b1
[63]	valid	1'b1

Address Region 2

The following figure shows the example configuration for Address Region 2.

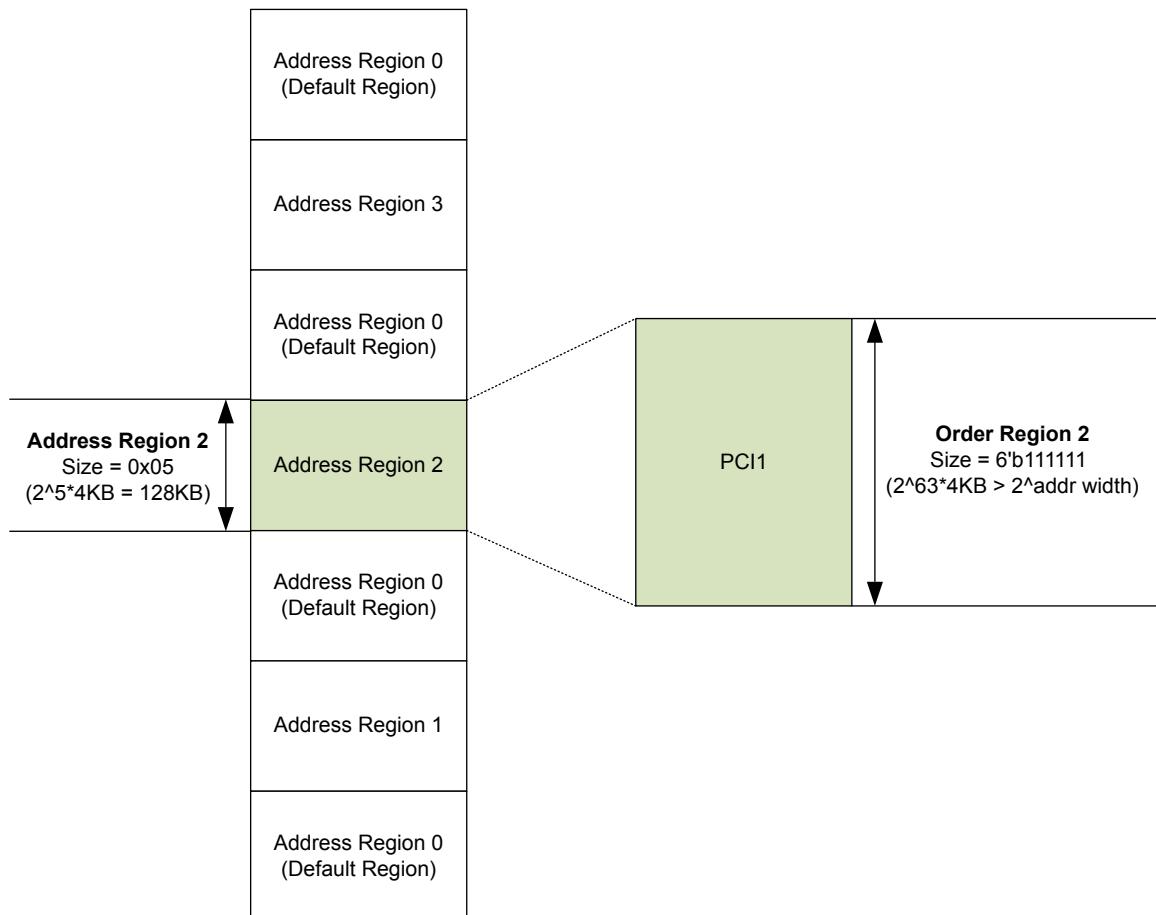


Figure 2-59 Address Region 2 configuration

Address Region 2 starts at base address `0x0000_0002_0000` and is 128KB in size. The Order Region 2 size ($2^{63} \times 4\text{KB}$) is configured to the maximum value (`6'b111111`), so Address Region 2 is considered as one Order Region. PCI1 occupies the entire Order Region, so all PCI1 requests are ordered.

The following table shows the configured values for the Address Region 2 Configuration Register, `por_hni_sam_addrregion2_cfg`.

Table 2-50 Address Region 2 configuration

Bits	Field name	Configured value
[5:0]	order_reg_size	<code>6'b111111</code>
[15:10]	addr_region_size	<code>6'h5</code>
[55:20]	base_addr	<code>36'h0000_0002_0</code>
[55:16]	base_addr	<code>40'h0000_0002_0</code>
[58]	physical_mem_en	<code>1'b0</code>
[59]	ser_all_wr	<code>1'b0</code>
[60]	ser_devne_wr	<code>1'b0</code>
[61]	pos_early_rdack_en	<code>1'b1</code>
[62]	pos_early_wr_comp_en	<code>1'b1</code>
[63]	valid	<code>1'b1</code>

Address Region 3

The following figure shows the example configuration for Address Region 3.

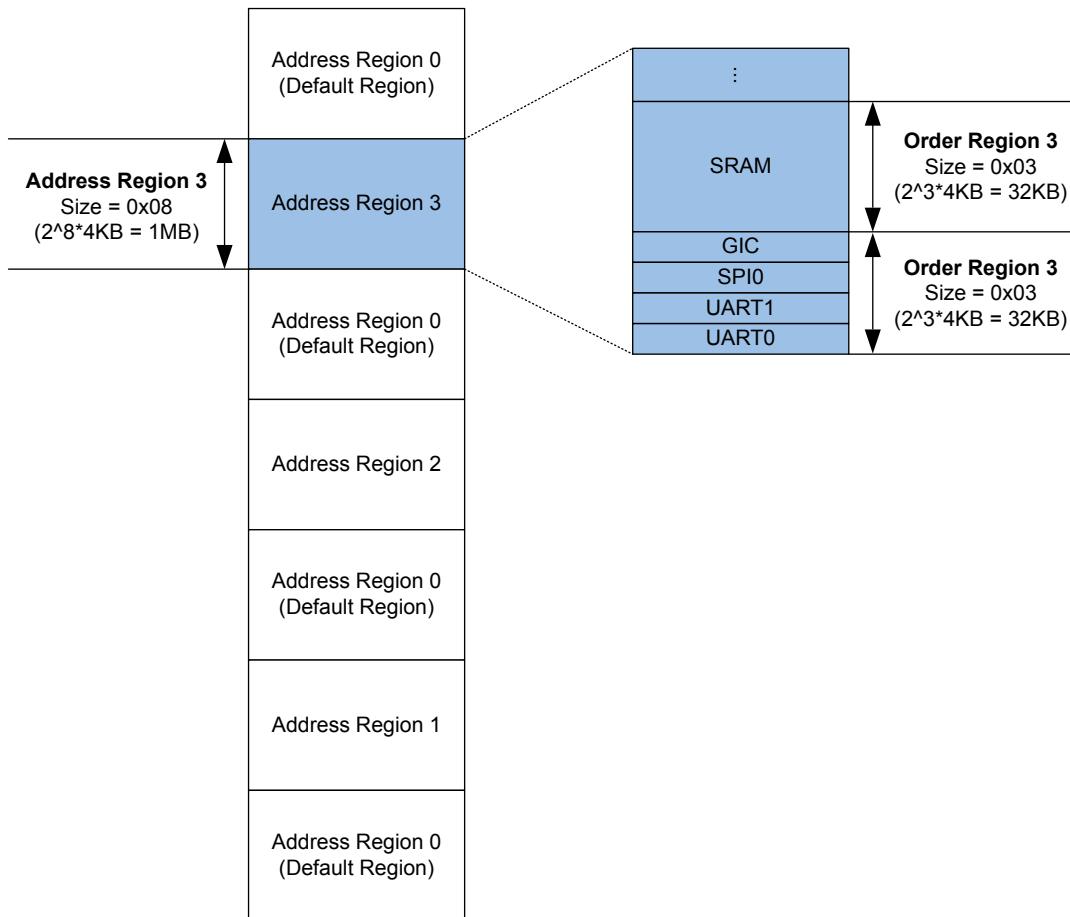


Figure 2-60 Address Region 3 configuration

Address Region 3 starts at base address `0x0000_0020_0000` and is 1MB in size. The Order Region 3 size of 32KB is less than the Address Region 3 size of 1MB, resulting in a total of 32 Order Regions. GIC, SPI0, UART1, and UART0 map to one Order Region, therefore all requests to these peripherals are ordered. SRAM also maps to one Order Region, therefore all requests to SRAM are ordered. Since SRAM maps to a separate Order Region from GIC, SPI0, UART1, and UART0, requests to SRAM and requests to GIC, SPI0, UART1, and UART0 are not ordered.

The following table shows the configured values for the Address Region 3 Configuration Register, `por_hni_sam_addrregion3_cfg`.

Table 2-51 Address Region 3 configuration

Bits	Field name	Configured value
[5:0]	order_reg_size	6'h3
[15:10]	addr_region_size	6'h8
[55:20]	base_addr	36'h0000_0020_0
[55:16]	base_addr	40'h0000_0020_0
[58]	physical_mem_en	1'b0
[59]	ser_all_wr	1'b0

Table 2-51 Address Region 3 configuration (continued)

Bits	Field name	Configured value
[60]	ser_devne_wr	1'b0
[61]	pos_early_rdack_en	1'b1
[62]	pos_early_wr_comp_en	1'b1
[63]	valid	1'b1

— Note —

In HN-P, SAM programming does not apply to requests from PCIe RN-Is or PCIe RN-Ds. PCIe RN-Is and PCIe RN-Ds are designated by the `pcie_mstr_present` configuration bit in the RN-I or RN-D node. Requests from these node types always assume that traffic is directed to endpoint memory space. The processing of requests from these sources is optimized according to PCIe ordering rules.

2.4.10 GIC communication over AXI4-Stream ports

CMN-700 supports optional master/slave *AXI4-Stream* (A4S) ports on RN-I, RN-D, and MXP RN-F ports for communication between a *Generic Interrupt Controller* (GIC) and CPUs. Certain requirements apply to the A4S routing and signaling.

CMN-700 also supports transmission of GIC information across CCIX links for CML SMP configurations.

More system-level information is available in the *Arm® Neoverse™ N1 hyperscale reference design GIC-600 Integration using CMN-600 AXI4-Stream Interfaces White Paper* on request.

A4S routing

The A4S ports are addressed according to Logical ID, which are assigned sequentially from 0 to the number of A4S ports. To send a packet from one A4S port to the destination A4S port, assign the TDEST to one of the following Logical IDs:

- The Logical ID of the target A4S port
- The Logical ID of the CXRH for GIC traffic targeting the other chip

The discovery process returns the number of A4S ports and Logical ID information for each A4S port. This information is collected by reading corresponding RN-I, RN-D, and XP unit information registers. For more information about the discovery process, see [2.5 Discovery on page 2-163](#).

Other requirements

- The **PUB_DESTID** associated with the GICD A4S port must drive the **CXRH_GICD_DESTID** input strap. The **PUB_DESTID** value is included in the CMN ID-mapping file that is created during the IP rendering process. For more information, see the *Arm® Neoverse™ CMN-700 Coherent Mesh Network Configuration and Integration Manual*, which is only available to licensees.
- GICD drives the CMN-700 **RXA4STRI[7:0]** input (8 MSB bits of **GICD_ICDRTDEST**), indicating the CCIX link of the target chip. GICD also drives the **RXA4STDEST[7:0]** (8 LSB bits of **GICD_ICDRTDEST**) of CXRH for CML SMP configurations.

— Note —

This requirement only applies to 2-chip configurations. Contact *Arm®* for information on three or more chip configurations.

- The A4S master must assert **valid** irrespective of **ready** state to transmit data.

2.4.11 Default XY routing behavior

By default, CMN-700 uses an XY routing algorithm to decide which direction to route flits within the mesh. At each MXP, the XID and YID values of the target MXP and the current MXP are compared to determine the routing direction.

Routing directions are referred to by the mesh port that the MXP routes the flit through. For example, if the MXP routes the flit northwards, then the flit is sent through the north mesh port.

If there is a mismatch between the target MXP XID and the current MXP XID, then the MXP uses the following rule to decide the routing direction:

- If target MXP XID > current MXP XID, then route eastwards.
- Otherwise, route westwards.

If the target MXP XID and the current MXP XID match, then the flit routing components are compared against the YID of the MXP. If YIDs do not match, then the MXP uses the following rule to decide the routing direction:

- If target MXP YID > current MXP YID, then route northwards.
- Otherwise, route southwards.

If the target MXP XID and YID match the current MXP XID and YID, then the flit has reached the target MXP. At this point, the flit is downloaded to the target device.

The following figure shows the default XY routing flow.

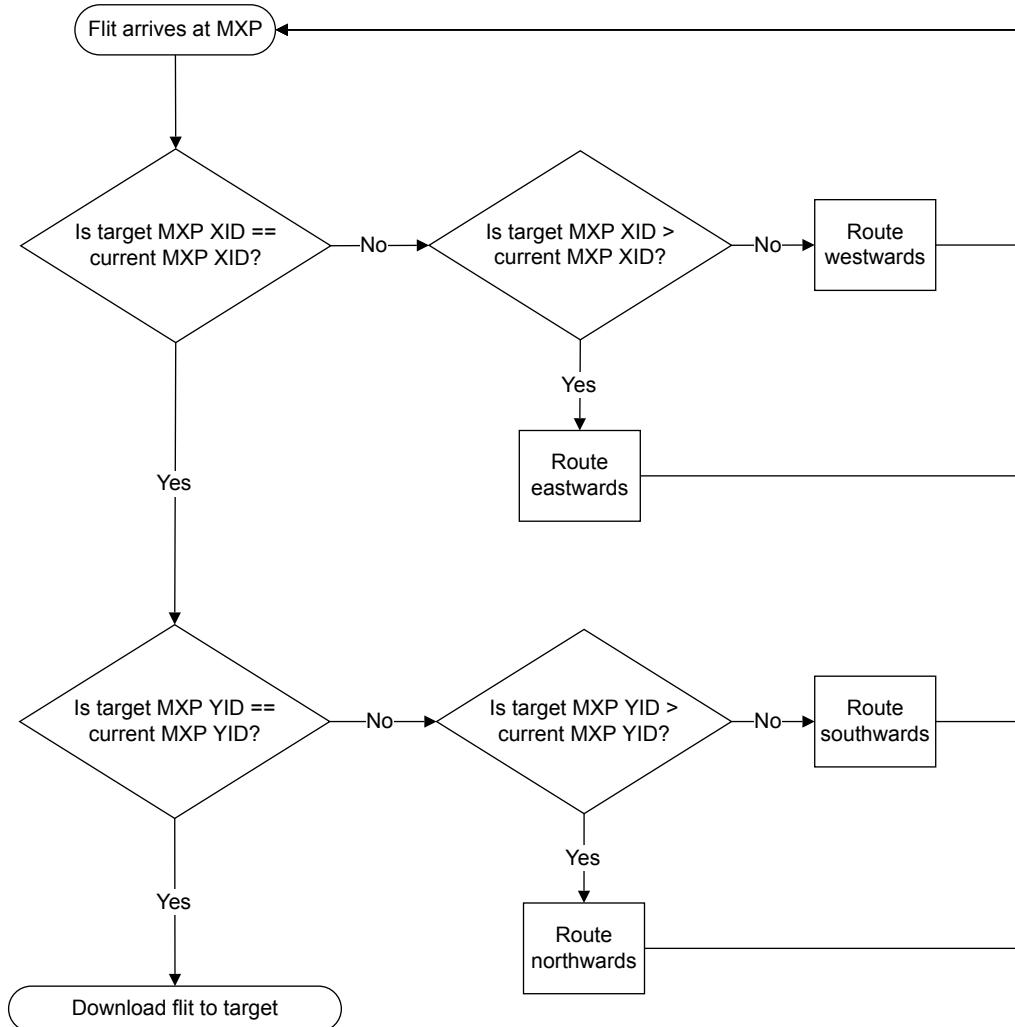


Figure 2-61 Default XY routing flow

You can configure CMN-700 to override the default XY routing pattern for specific source-target pairs in the mesh. For more information about this feature, see [2.4.12 Non-XY routing on page 2-148](#).

2.4.12 Non-XY routing

You can configure up to 16 XP pairs in the CMN-700 mesh to route CHI traffic against the default XY routing algorithm. Non-XY routing improves the efficiency of traffic flow by reducing hotspots in the mesh layout.

By default, CMN-700 uses an XY routing mechanism to route flits through the mesh. For more information about the default XY routing mechanism, see [2.4.11 Default XY routing behavior on page 2-147](#).

You can configure any source-target pair of XPs in your mesh configuration to use non-XY routing, up to a maximum of 16 pairs.

You enable this optional feature using the `XY_OVERRIDE_CNT` parameter, which supports values 0, 2, 4, 8, or 16. The value represents the number of source-target pairs which use non-XY routing. To define the

non-XY routing XP pairs and their behavior, you can program the por_mxp_xy_override_sel_* registers at boot.

Based on an identified hotspot, one or two XPs are selected for non-XY routing. The first XP is the point of XY route override, which overrides the XY route for the flit while still honoring the XY algorithm. The second XP is the point of YX turn where the flit is routed in the Y direction.

This feature only applies to the CHI channels, REQ, RSP, DAT, and SNP, not to the PUB channel.

Configuring non-XY routing behavior

A boot-programmable static *Lookup Table* (LUT) in each XP controls non-XY routing.

You enable support for this feature by setting the XY_OVERRIDE_CNT parameter. For more information, see [2.4.12 Non-XY routing on page 2-148](#) and [1.5.2 Mesh sizing and top-level configuration on page 1-21](#).

Eight 64-bit boot-programmable registers control the non-XY routing feature (por_mxp_xy_override_sel_* registers). These registers support override of the route paths for up to 16 source-target XP pairs.

The contents of the por_mxp_xy_override_sel_* registers represent a static LUT. The following table shows the format of each entry in the LUT.

Table 2-52 LUT entry format

Field	Description
<SRCID>	The source ID of the source-target pair that is enabled for XY override.
<TGTID>	The target ID of the source-target pair that is enabled for XY override.
CAL TGT PRESENT	CAL TGT Presence Indication for XY Route Override of all devices behind CAL.
YX turn enable	Allows YX turn in the XP.
XY route override	Enables flit XY route override in the XP.

The following table shows the structure of a single non-XY routing register.

Table 2-53 por_mxp_xy_override_sel_* structure

Bitfield	Name
[63]	VALID
[62:59]	Reserved
[58:48]	srcid_1
[47]	Reserved
[46:36]	tgtid_1
[35]	Reserved
[34]	cal_tgt_present_1
[33]	yx_turn_enable_1
[32]	xy_override_enable_1
[31:27]	Reserved
[26:16]	srcid_0
[15]	Reserved
[14:4]	tgtid_0
[3]	Reserved

Table 2-53 por_mxp_xy_override_sel_* structure (continued)

Bitfield	Name
[2]	cal_tgt_present_0
[1]	yx_turn_enable_0
[0]	xy_override_enable_0

When routing flits between XPs, the XP compares the <SRCID> and <TGTID> flit fields against the entries in this LUT. This comparison, along with the YX turn enable and XY route override values for each XP, identify the route for the flit to take.

For the specific programming sequence to set up the LUT, see [3.4.5 Program non-XY routing registers on page 3-1243](#).

Rules for avoiding deadlocks in non-XY routing

You must follow various rules to ensure that the non-XY routing implementation is free of deadlocks.

In the default XY routing scheme, the following turns are forbidden:

- S → E
- N → W
- S → W
- N → E

For non-XY routing, these turns are allowed, but you must apply the following rules to avoid deadlocks. xj or xi represents the XID value of an XP, and yj or yi represents the YID value of an XP.

- If N → W turn is allowed at $XP_{xi,yi}$, then S → E turn is disallowed at every $XP_{xj,yj}$ where ($xj < xi$) and ($yj < yi$).
- If S → E turn is allowed at $XP_{xi,yi}$, then N → W turn is disallowed at every $XP_{xj,yj}$ where ($xj > xi$) and ($yj > yi$).
- If N → E turn is allowed at $XP_{xi,yi}$, then S → W turn is disallowed at every $XP_{xj,yj}$ where ($xj > xi$) and ($yj < yi$).
- If S → W turn is allowed at $XP_{xi,yi}$, then N → E turn is disallowed at every $XP_{xj,yj}$ where ($xj < xi$) and ($yj > yi$).

Non-XY routing examples

As an example, consider a flit that is uploaded from decimal source NodeID 40 and targets decimal NodeID 124 on a 4x4 mesh configuration.

The following figure shows the default routing of the flit without non-XY routing.

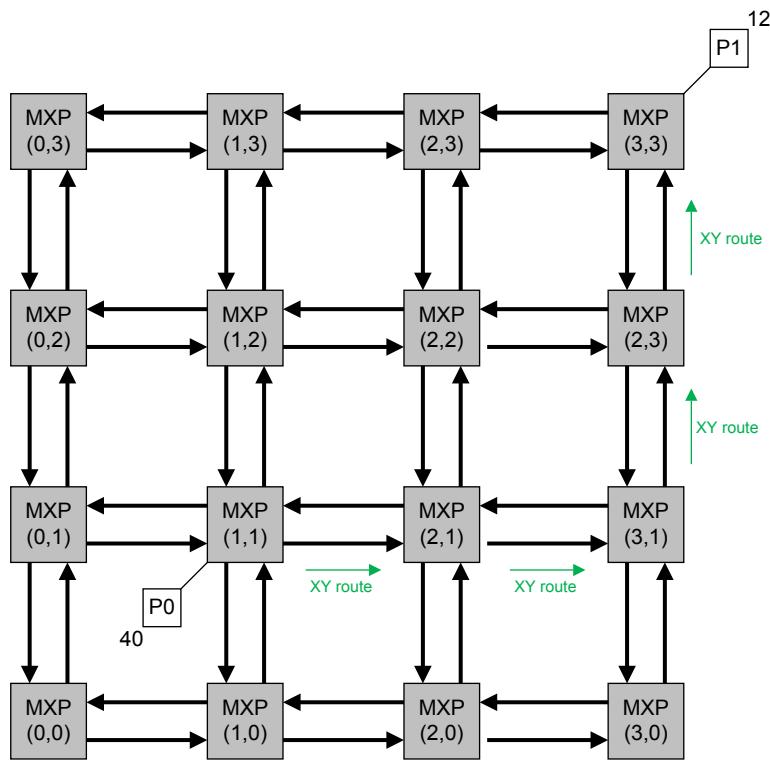


Figure 2-62 Default XY routing example

According to the standard XY routing algorithm, the flit follows the following route:

1. MXPs route the flit in the east direction until the flit reaches MXP_(3,1).
2. MXPs route the flit in the north direction to MXP_(3,3), where the target node downloads the flit.

XY override enabled, YX turn disabled

The following figure shows the default routing of a flit and the XY override route.

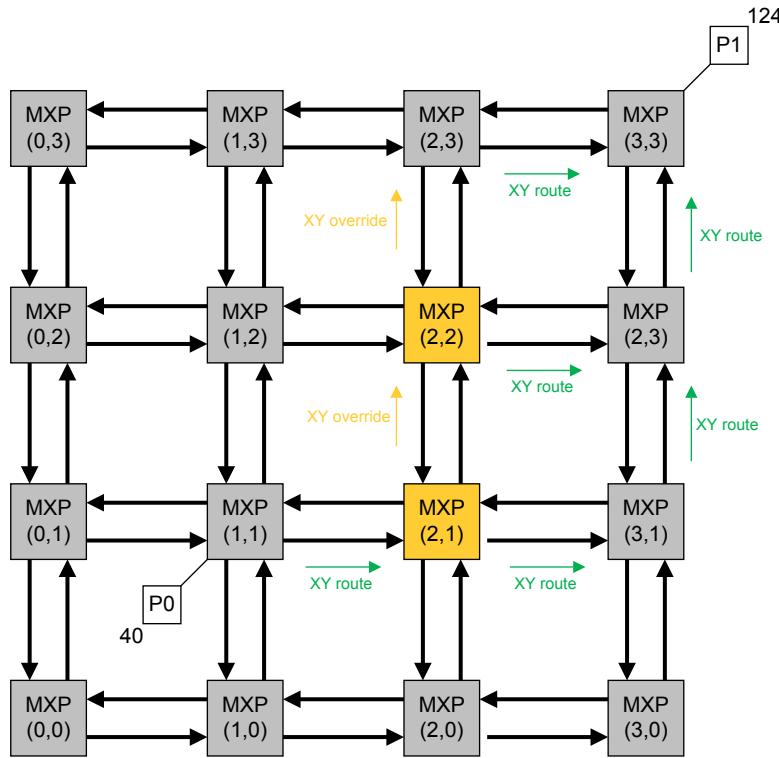


Figure 2-63 XY override enabled and YX turn disabled routing example

In the example, the non-XY routing registers in MXP_(2,1) and MXP_(2,2) are configured to override the XY route for a set of source-target pairs. This set includes NodeID40 and NodeID124, so the flit follows the following route:

1. MXP_(1,1) routes the flit in the east direction.
2. MXP_(2,1) and MXP_(2,2) route the flit in the north direction, since their configuration has XY override enabled.
3. There is no override set in MXP_(2,3). Therefore, the MXP routes the flit in the east direction according to the default XY routing algorithm.
4. At MXP_(3,3), the flit has reached its destination, and the target node downloads the flit.

If the XY override option is enabled and YX turn option is disabled, the following assumptions and constraints apply to the routing algorithm:

- If target MXP YID \geq current MXP YID and a northern mesh port is present, then route northwards.
- If target MXP YID $<$ current MXP YID and a southern mesh port is present, then route southwards.
- If target MXP YID == current MXP YID, a southern mesh port is present, and a northern mesh port is absent, then route southwards.
- Otherwise follow the default XY routing algorithm.

Both XY override and YX turn enabled

The following figure shows the default routing of a flit and an example XY override and YX turn enabled route.

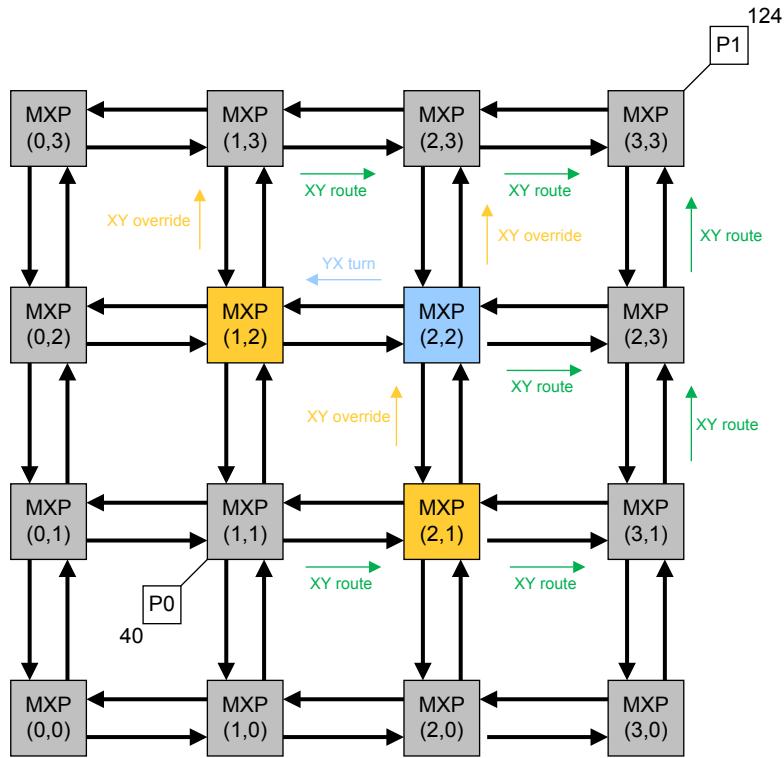


Figure 2-64 XY override and YX turn enabled routing example

In the example, to enable non-XY routing, the following configurations have been made for a set of source-target pairs, including NodeID40 and NodeID124:

- Non-XY routing registers in MXP_(1,2) and MXP_(2,1) are configured to override the XY route.
- Non-XY routing registers in MXP_(2,2) are configured to override the XY route and enable YX turns.

Therefore, the flit follows the following route:

1. MXP_(1,1) routes the flit in the east direction.
2. MXP_(2,1) routes the flit in the north direction, since its configuration has XY override enabled.
3. MXP_(2,2) routes the flit in the west direction, since its configuration has XY override and YX turn enabled.
4. MXP_(1,2) routes the flit in the north direction, since its configuration has XY override enabled.
5. There is no override set in MXP_(1,3) and MXP_(2,3). Therefore, the flit is routed in the east direction according to the default XY routing algorithm.
6. At MXP_(3,3), the flit has reached its destination, and the target node downloads the flit.

If XY route override and YX turn are enabled, the following assumptions and constraints apply to the routing algorithm:

- If target MXP XID \leq current MXP XID and an eastern mesh port is present, then route eastwards.
- If target MXP XID $>$ current MXP XID and a western mesh port is present, then route westwards.
- If target MXP XID == current MXP XID, a western mesh port is present, and an eastern mesh port is absent, then route westwards.
- Otherwise follow the default XY routing algorithm.

2.4.13 Extended CCIX Requesting Agent ID mechanism for up to 512 RN-Fs

CMN-700 CCIX RAs are identified using a globally unique 10-bit *Requesting Agent ID* (RAID). This ID is made up of a 6-bit base ID and a 4-bit expanded ID.

The base ID component of the RAID is based on the CCIX Agent ID, as defined in the *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*. The 4-bit expansion to the RAID enables CMN-700 to support up to 512 RAs in a system, while still being compatible with CCIX devices. To be compliant with the CCIX specification, all uniquely identifiable agents on a device must be assigned unique IDs from the 6-bit base ID space.

For CMN-700 RAIDs, the 6-bit base ID must be unique to each chip. For example, consider a configuration where one chip is assigned the base ID[5:0] = 0x4. In this case, no other chip in the system can have RAIDs with the base ID[5:0] = 0x4.

A mixed system can comprise devices containing CMN-700 instances and devices that do not contain CMN-700. However, devices that do not contain CMN-700 are compliant with revision 1.1 of the CCIX specification. Because the CMN-700 RAID is larger than the CCIX Agent ID, some assignment rules apply to these mixed systems. You must apply the following rules when assigning IDs to devices in a mixed system:

1. Assign a 6-bit unique CCIX Agent ID to each uniquely identifiable agent on all non-CMN-700 CCIX devices. For more information about assigning CCIX Agent IDs, including further rules and requirements that are not described in this document, see the *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*. The 4-bit expanded ID space cannot be used in the system for these non-CMN-700 CCIX devices.
2. Identify all the CMN-700 agents that can communicate with the preceding CCIX agents and assign them a 6-bit unique ID. The expanded ID for these agents must be 0b0000 (0x0).
3. Assign all remaining CMN-700 agents an ID from the remaining 10-bit ID space.

Example 2-9 RAID assignment in an example mixed CML system

Consider a system configuration containing two CMN-700 chips, P0 and P1, and two accelerator chips, A0 and A1. The chips have the following characteristics:

- P0** Chip is built using CMN-700, supports 10-bit RAID, and contains 128 RAs.
- P1** Chip is built using CMN-700, supports 10-bit RAID, and contains 256 RAs.
- A0** Chip is compliant with revision 1.1 of the CCIX specification, supports 6-bit CCIX Agent ID, and contains 16 RAs.
- A1** Chip is compliant with revision 1.1 of the CCIX specification, supports 6-bit CCIX Agent ID, and contains 8 RAs.

The following figure shows this example CML system.

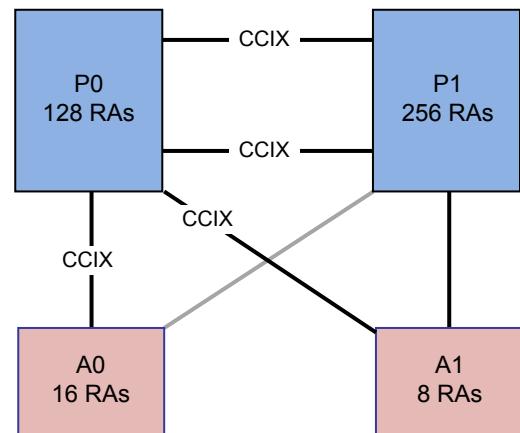


Figure 2-65 Example CML system

RAIDs across P0, P1, A0, and A1, are assigned according to the following scheme:

1. A0 and A1 have 24 RAs in total. Therefore, they can be assigned RAIDs 0-23.
2. P0 has 128 RAs. P0 requires a minimum of eight unique base ID values because the 4-bit expanded ID space provides 16 unique RAIDs for each 6-bit base ID value. To find the number of unique base ID values that are required for a CMN-700 chip, divide the number of RAs by 16, or 2^4 .
3. P1 has 256 RAs. According to the calculation in the preceding step, P1 needs a minimum of 16 unique IDs from the base ID space.
4. Identify all the agents on P0 and P1 that can communicate with A0 and A1. The number of these agents can affect the final number of unique base IDs required for P0 and P1. For example, 16 RAs from P0 and 8 RAs from P1 can communicate with A0 and A1.
5. Because P0 has 16 RAs that can communicate with A0 and A1, a total of 16 unique base IDs must be used for these RAs.
 - Assign IDs 24-39 to the RAs on P0 that can communicate with A0 and A1.
6. P1 has eight RAs that can communicate with A0 and A1, but the minimum number of unique base IDs that are required is 16, according to the number of RAs on the chip.
 - Assign IDs 40-55 to RAs on P1. Eight of these IDs must be assigned to RAs that can communicate with A0 and A1. The other eight can be assigned to any of the RAs on P1.
7. All remaining RAs which have not been assigned an ID on P0 and P1 can be assigned IDs from the remaining 10-bit ID space.

For the preceding example system, LDIDs in the CMN-700 device P0 are assigned in the following way:

1. 128 RAs on P0 are assigned LDIDs from 0-127.
2. 256 RAs from P1 are assigned LDIDs from 128-383.
3. 16 RAs from A0 are assigned LDIDs from 384-399.
4. 8 RAs from A1 are assigned LDIDs from 400-407.

LDIDs in the CMN-700 device P1 are assigned in the following way:

1. 256 RAs on P1 are assigned LDIDs from 0-255.
2. 128 RAs from P0 can be assigned LDIDs from 256-383.
3. 8 RAs from A1 can be assigned LDIDs from 384-391.
4. 16 RAs from A0 can be assigned LDIDs from 392-407.

For more information about LDIDs in cross-chip routing, see [2.4.14 Cross chip routing and ID mapping on page 2-155](#). You can also override the default LDIDs when SF clustered mode is enabled. For more information about SF clustered mode and LDID override functionality, see [4.2.11 Non-clustered and clustered mode for SF RN-F tracking on page 4-1283](#) and [4.2.13 Configuring clustered mode for SF tracking on page 4-1284](#).

Although HAs are not shown in the example, all the chips, including A0 and A1, could have HAs. HAIDs can be same as any of the RAIDs assigned to the chip and the HAIDs must be from the unique base ID pool. In other words, the 4-bit expanded ID must be `0x0`.

CMN-700 RN-Is or RN-Ds that communicate on a non-SMP link must be given a unique 6-bit ID. This ID must not be the same as the RAID of any of the RAs on the chip.

CMN-700 RN-Is or RN-Ds that communicate on an SMP link can have the same RAID as any RA on the chip.

2.4.14 Cross chip routing and ID mapping

IDs are generated and used to route protocol messages across multiple chips.

This section covers ID generation and the methods that are used to route CCIX and CML_SMP protocol messages across multiple chips. The following acronyms are used in this section:

- *Request Agent ID* (RAID)
- *Home Agent ID* (HAID)
- *Logical Device ID* (LDID)

By default, LDIDs are uniquely assigned within a device type. For example, RN-Fs in the system could be assigned LDIDs 0-n, while RN-Is could be assigned LDIDs 0-m, and RN-Ds could be assigned LDIDs 0-k.

The following rules apply to configurations with HN-F SF clustered mode enabled:

- RAID usage:
 - Is confined to CML gateway devices only.
 - All local and remote components that are visible to CMN-700 use and operate on sequentially assigned LDIDs. CXG/CCG devices bidirectionally map each RAID to an LDID.
- RN-F default LDID assignment:
 - Local RN-Fs are assigned LDIDs from 0-n, sequentially.
 - Remote RN-Fs must be assigned LDIDs n+1 and above by the discovery software.

If HN-F SF clustered mode is enabled, the preceding rules are not needed. In this mode, local and remote RN-Fs can be assigned LDIDs from the full ID space to meet the clustering need. The **LDID_WIDTH** parameter determines the full ID space. To maximize SF efficiency, assigned LDIDs must be sequential and the LDID space must not have any holes.

The following figure shows a basic multi-chip block diagram.

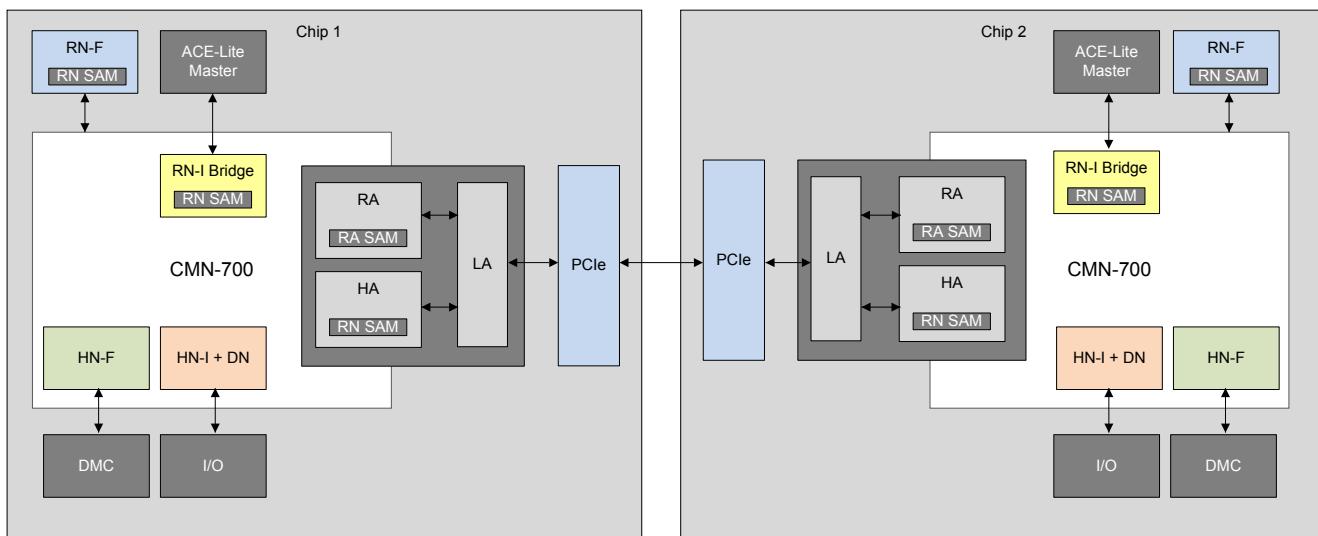


Figure 2-66 Multi-chip block diagram

Request from an RN-F to a remote HN-F

The following figure shows all IDs generated and used to route a request from a local RN-F on Chip 1 to a remote HN-F on Chip 2.

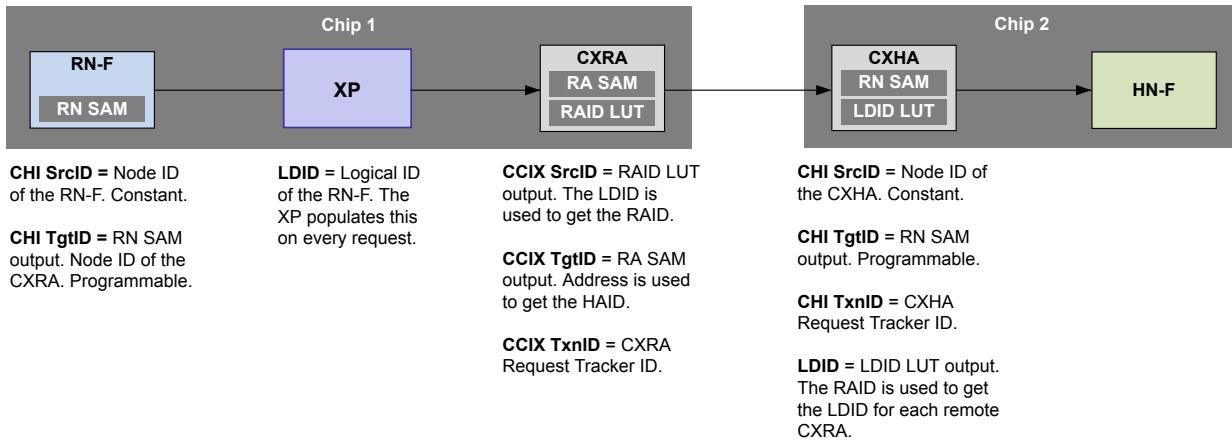


Figure 2-67 RN-F to remote HN-F IDs

The flow for this process is as follows:

- The RN-F looks up the programmable RN SAM to populate the CHI target ID on a request.
- The XP populates the RN-F LDID on this request.
- When HN-F SF clustering is disabled, the LDIDs for local RN-Fs must not be changed. The build-time LDID assignments are discovered by reading any one of the HN-F `por_hnf_rn_cluster<X>_physid` registers. A few cycles after reset, these registers are prepopulated with the LDIDs for local RN-Fs within that chip.
- If HN-F SF clustered mode is enabled:
 - The local RN-F LDID in `por_mxp_p[0-5]_ldid_override` register can be programmed to match the clustering requirements at each RN-F port. In clustered mode, the local RN-F LDIDs are not pre-programmed in the physical ID registers of the HN-F out of reset. Therefore, you must explicitly program these registers to suit the clustering needs.
 - When HN-F SF clustered mode is enabled, CMN-700 allows override of the LDID of each RN-F. This LDID must be programmed in the [`por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127`](#) registers of the CXRA. Also, each override value must match the `por_mxp_p[0-5]_ldid_override` register value of the corresponding RN-F.
 - If HN-F SF clustered mode is enabled, there must be at least two cluster groups. In other words, you cannot cluster all RN-Fs into a single cluster group.
 - If HN-F SF clustered mode is enabled, and there is a single RN-F in the cluster, then it must always use the lowest device ID in that cluster. Consider a four-way clustering, where the device ID fields are `0b00`, `0b01`, `0b10`, and `0b11`. In this configuration, you cannot use device ID values of `0b01`, `0b10`, or `0b11` unless `0b00` is also in use.
- `por_mxp_device_port_connect_ldid_info_p[0-5]` captures the default LDID values assigned to the RN-Fs that are connected to the respective device port.
- The `por_hnf_rn_cluster_*_physid_reg*` registers also contain fields to program the source type for each RN-F in the system. For all local RN-Fs, the source type must be programmed to the appropriate CHI protocol issue that the RN-F supports. For all remote RN-Fs, the source type must be programmed to `0b1100` (CHI-E) as CXHA is a proxy for all remote RN-Fs.
- RN-Is and RN-Ds are internal to CMN-700 and get their logical ID assigned during CMN-700 generation. The RN-I or RN-D sends this LDID on every request.
- The CXRA contains programmable lookup tables, RAID LUTs, for each class of local RN (RN-F, RN-I, and RN-D). CCIX or CML_SMP discovery software discovers all local RN-Fs, RN-Ds, and RN-Is, and programs their corresponding RAIDs in these LUTs. The LDID of the incoming request is used to look up these RAID LUTs and determine the CCIX RAID. CXRA also has CCIX RA SAM.

This CML RA SAM is used to generate the HAID. This HAID is used as the target ID to route the CML request message.

- The build-time LDID assignment can be discovered by reading any of the por_{cxg,ccg}_ra_rnf_ldid_to_nodeid_reg registers.

The following figure shows the programmable registers during CML Discovery.

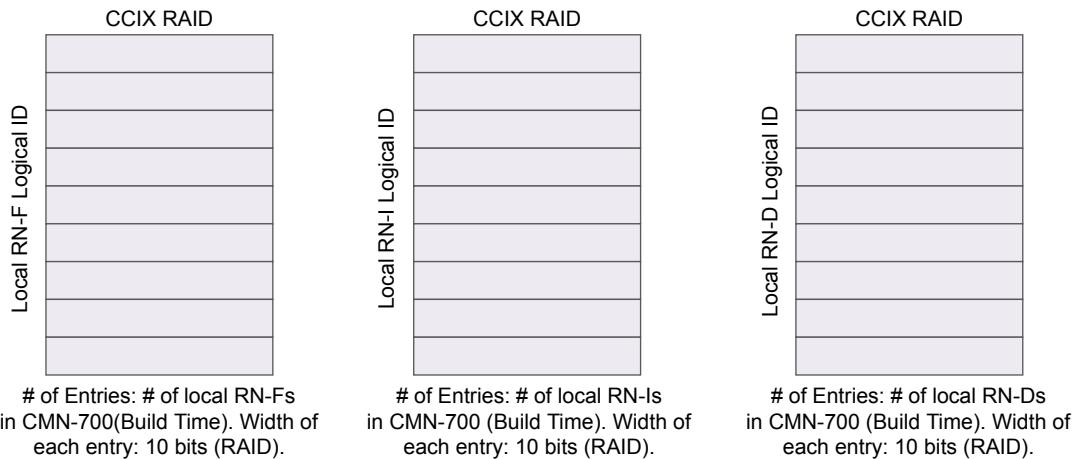


Figure 2-68 Programmable registers during CML Discovery

The CXHA contains a programmable register to program the local LDID for each remote CML RAID that communicates with local CHI HNs on a given chip or socket. Each entry in the programmable register also contains an RN-F bit to identify whether the remote CXRA is a caching agent (RN-F) or not. If RN-I and RN-F have the same RAID, then you must only fill the RN-F details in the entry. HN-Fs on the local chip use this LDID to track a line in its SF. Therefore a unique LDID assignment is required for each remote requesting caching agent. These unique LDIDs must not overlap with LDIDs assigned to local RN-Fs. It is assumed that these IDs are assigned after CML Discovery is complete. For example, all the CXRAs are discovered and assigned an RAID.

The following figure shows the programmable register for RAID to LDID during CML Discovery.

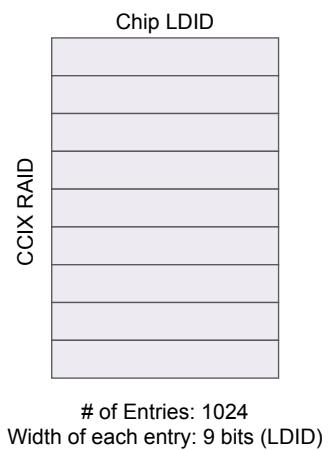


Figure 2-69 RAID to LDID during CML Discovery

With the LDID passed to HN-F in all CHI REQ flits, HN-F uses this LDID as the true logical ID for SF tracking purposes. HN-F uses a logical ID vector in the SF. Depending on whether SF clustered mode is enabled or not, the LDIDs can either be uniquely tracked in the SF or multiple LDIDs can be aliased to a single logical ID. The total number of bits in the SF vector is calculated based on configuration parameters. You can also make the vector larger using the NUM_ADD_SF_VECTOR configuration parameter.

The following figure shows all generated IDs used to route a response from a remote HN-F on Chip 2 to an RN-F on Chip 1.

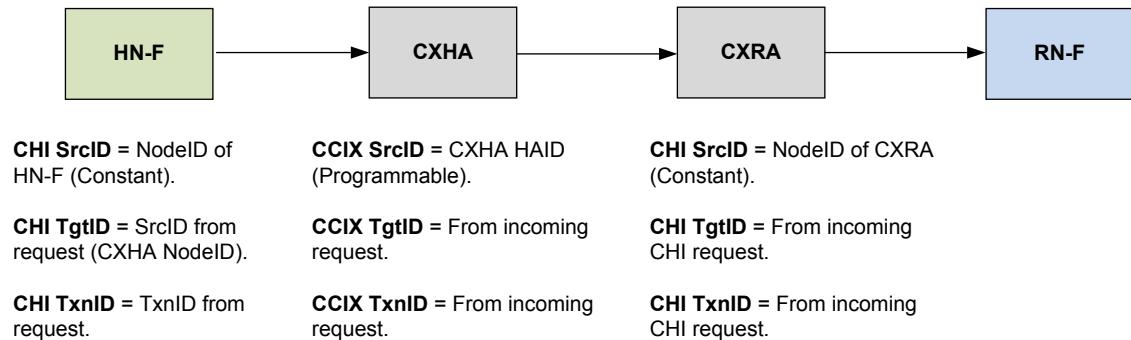


Figure 2-70 Remote HN-F to RN-F IDs

The following figure shows the flow of a snoop from an HN-F to a remote RN-F.

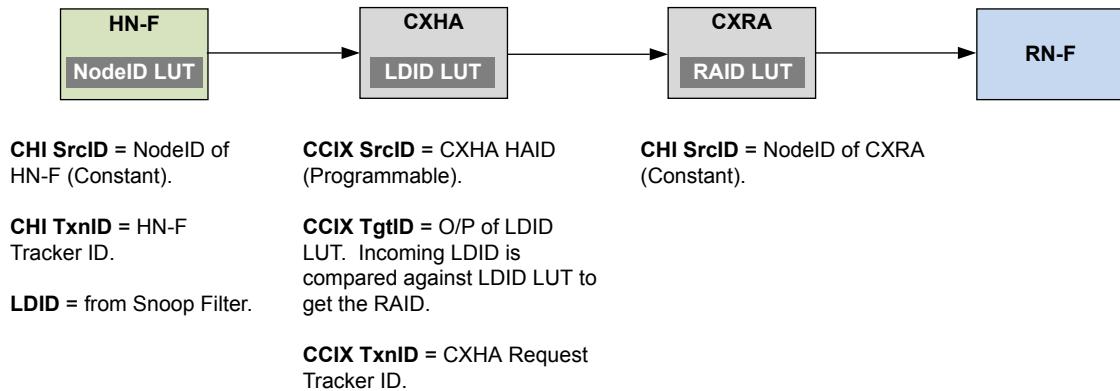


Figure 2-71 Snoop from HN-F to remote RN-F

The HN-F contains the following programmable LUT to program the CXHA node ID of each remote caching agent. HN-F uses the unique LDID from the snoop vector to look up the physical CHI node ID of the CXHA where snoops are sent to. The following table shows an example programming where logical IDs 0-7 are assigned to the local RN-Fs. The logical IDs 8-15 are assigned to the remote RN-Fs. Software assigns these IDs during the Discovery process.

Table 2-54 Example program

Logical ID as index	HN-F programmable register	
	CHI node ID	ID valid
0	Local RN-F 0	1
1	Local RN-F 1	1
2	Local RN-F 2	1
...
7	Local RN-F 7	1
8	CXHA	1

Table 2-54 Example program (continued)

Logical ID as index	HN-F programmable register	
	CHI node ID	ID valid
9	CXHA	1
...
15	CXHA	1
16	Not programmed	0
...	Not programmed	0
n	Not programmed	0

The CXHA uses the LDID from incoming CHI snoop to perform a content match against the entries of programmable CML RAID to local LDID LUT. This content match results in the CML RAID sending a CCIX snoop, as the following figure shows.

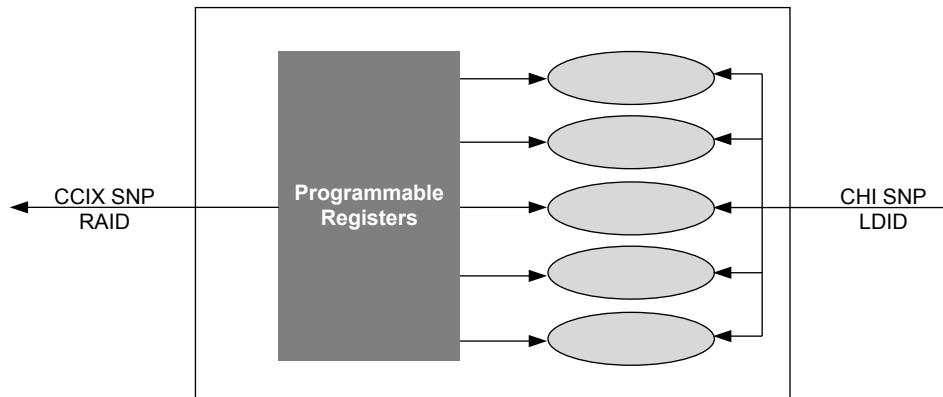


Figure 2-72 CHI SNP LDID to CML SNP RAID flow

The following figure shows the detailed flow of a CHI SNP LDID to CML SNP RAID conversion.

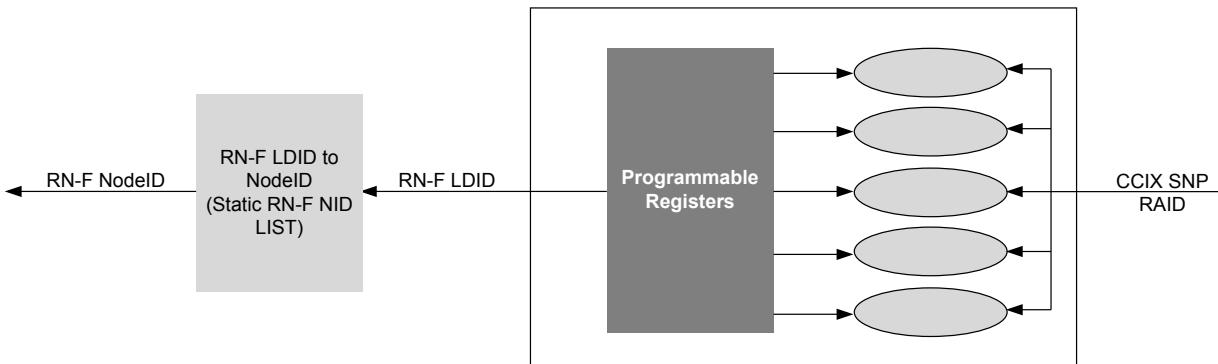


Figure 2-73 CML SNP RAID to CHI RN-F LDID flow

The following figure shows all generated IDs used to route a snoop response from a remote RN-F on Chip 1 to an HN-F on Chip 2.

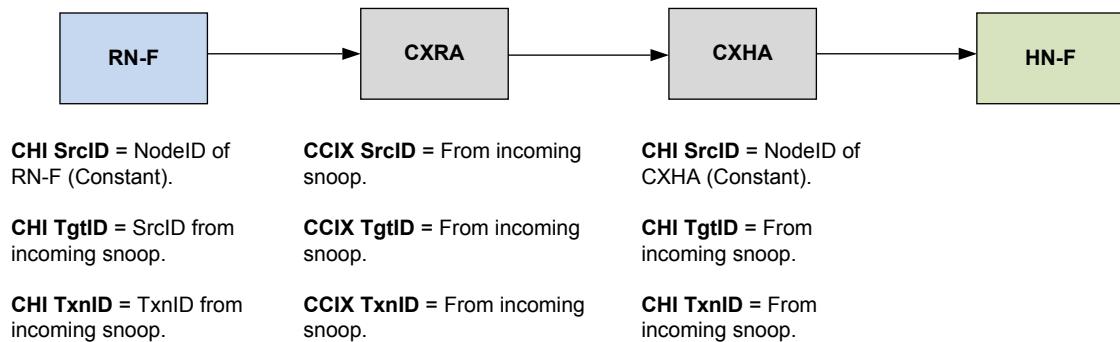


Figure 2-74 Remote RN-F to HN-F with all IDs generated

2.4.15 CCIX1.1 port-to-port forwarding

CMN-700 supports CML configurations with multiple CXS connections. CMN-700 can pass transactions from one CXS connection to another through the mesh, eliminating the requirement for external switching logic.

The port-to-port forwarding feature allows the CMN-700 mesh to act as a bridge between two CCIX chips. Using this feature, you can create various multichip topologies using only CCIX links, including a CCIX daisy-chain and CCIX mesh.

The following figure shows a daisy-chain topology using port-to-port forwarding. Each chip can target traffic to every other chip in the chain.

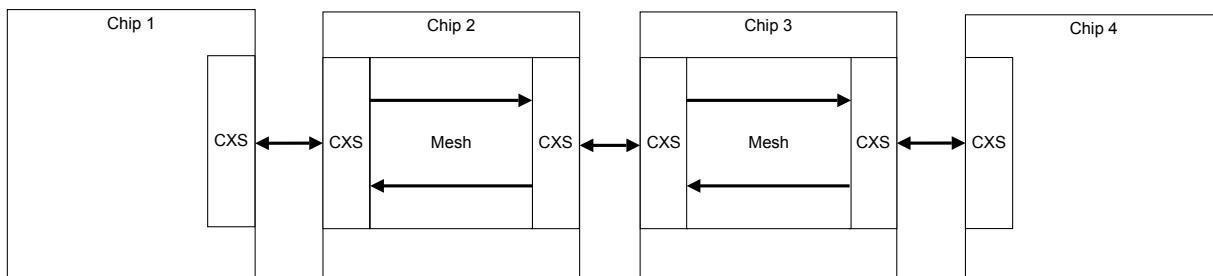


Figure 2-75 Four chip CCIX daisy chain configuration using port-to-port forwarding

The following figure shows a CCIX mesh topology using port-to-port forwarding. Each chip can target traffic to every other chip in the CCIX mesh.

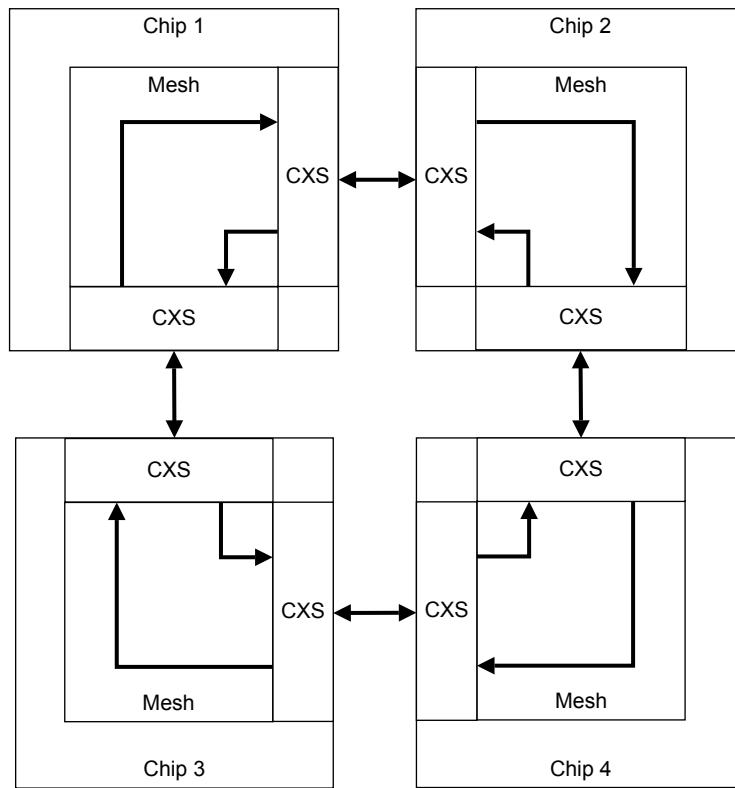


Figure 2-76 Four chip CCIX mesh configuration using port-to-port forwarding

For more information about programming the port-to-port forwarding feature, see [3.5.3 Program CML system to enable CCIX communication](#) on page 3-1260.

2.5 Discovery

Discovery is a software algorithm that is used to discover the configuration of CMN-700.

Software uses the discovery mechanism to identify:

- The CHI node ID and logical ID corresponding to all node types.

Note

The valid logical node types are DVM, Global CFG, DTC, HN-F, HNF_MPAM_S, HNF_MPAM_NS, HNP, HN-I, MTU, RN-D, RN SAM, RN-I, SBSX, and XP. There are other node types for additional functionality:

CML CXRA, CXHA, CXLA, CCRA, CCHA, CCLA, and CCLA_RNI.

- Whether a discovered node is internal or external to CMN-700.

The following figure shows an example configuration. In the example, after discovery, software should have enough information to know the location of the following components:

- Global configuration registers
- Configuration registers for each XP
- Configuration registers for the HN-F
- Configuration registers for the RN SAM corresponding to the RN-F

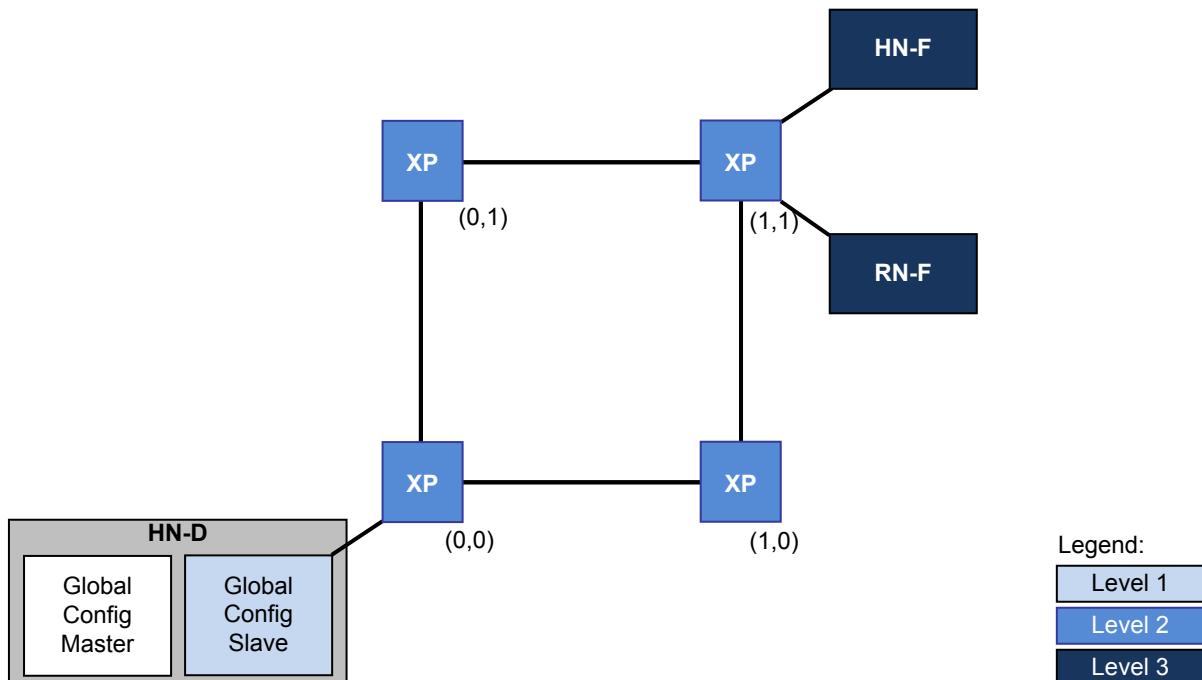


Figure 2-77 2 × 2 register tree example

This section contains the following subsections:

- [2.5.1 Configuration address space organization](#) on page 2-163.
- [2.5.2 Configuration register node structure](#) on page 2-165.
- [2.5.3 Child pointers](#) on page 2-167.
- [2.5.4 Discovery tree structure](#) on page 2-169.

2.5.1 Configuration address space organization

The way the configuration address space is organized depends on the system configuration. It is based on one system address, which is known as PERIPHBASE.

PERIPHBASE is the starting address of the range that all CMN-700 configuration registers are mapped to. For a CMN-700 system where both the X and Y dimensions are eight or less:

- This address must be aligned to 256MB.
- The maximum size of the address range is 256MB.

For a CMN-700 system where both the X and Y dimensions are nine or more:

- This address must be aligned to 1GB.
- The maximum size of the address range is 1GB.

Discovery determines specific addresses for individual system blocks that have **IMPLEMENTATION DEFINED** register spaces, as the following figure shows.

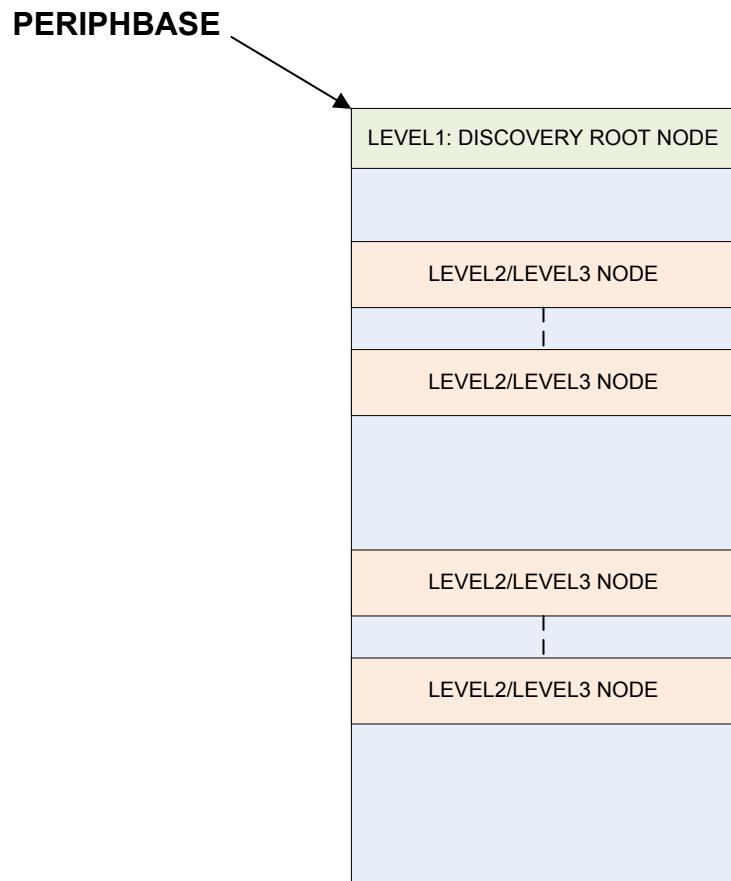


Figure 2-78 PERIPHBASE address map

CMN-700 supports 4B and 8B software-accessible registers. Register organization consists of software using 32-bit and 64-bit register reads.

All registers are organized into several register blocks as nodes. A node:

- Is a register block with the size of 64KB.
- Is associated with a logical block in the design.
- Has information and configuration for that block that is specific to the implementation.

The different types of nodes are:

General	Contains device information and has children.
Leaf	Contains device information, such as configuration data, but has no children.

Pure hierarchy Has children but contains no device information.

If a node has more than one child, the node provides:

- The number of children.
- A pointer to each child.

————— **Note** ————

You can also find the address offsets for each node and configuration register in the IP-XACT file that Socrates generates for your custom mesh. Socrates stores the IP-XACT file with the rendered RTL in your Socrates workspace.

2.5.2 Configuration register node structure

Read-only registers that are organized into several register blocks are referred to as nodes.

Nodes are aligned on 8B boundaries (64KB aligned). The required registers are:

Node Information register	Identifies the product or node type, and the CHI node ID.
Child Information register	Indicates the child count and offset for the first register containing child node pointers. These optional Child Pointer registers each use 8B.

————— **Important** ————

The Node Information and Child Information registers are at fixed offsets for all nodes.

The following figure shows the basic node structure.

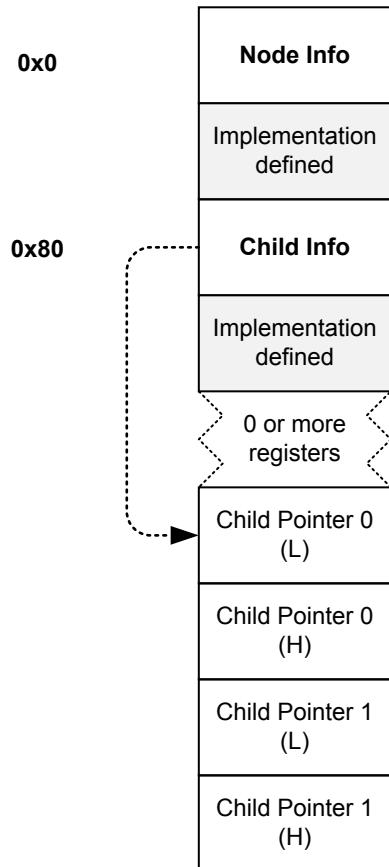


Figure 2-79 Basic node structure

The child_count field of the Child Information register indicates the number of children. This value also represents the number of functional units that are connected to the current unit on the next level of the discovery process.

The child_ptr_offset field of the Child Information register indicates the Child Pointer 0 register offset, in bytes, from the Node Information register address.

Important

For a leaf node (node with no children), the child_count and child_ptr_offset fields must be set to zero.

The following figure provides the node structure detail.

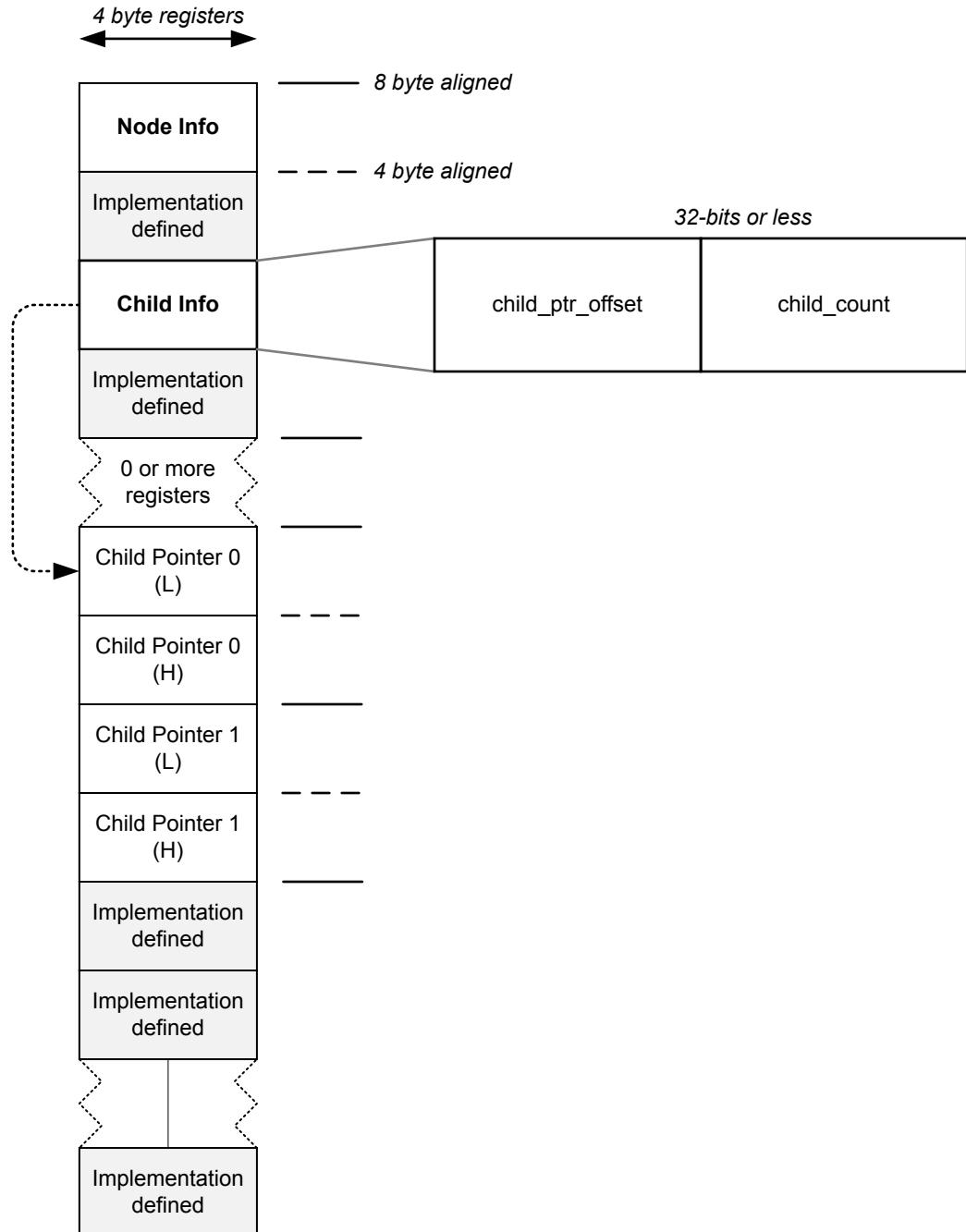


Figure 2-80 Node structure detail

The following table shows the supported node types and the corresponding node_type values in the Node Information register.

Table 2-55 node_type values

Node type	Value
Invalid	16'h0000
DVM	16'h0001
CFG	16'h0002
DTC	16'h0003
HN-I	16'h0004
HN-F	16'h0005
XP	16'h0006
SBSX	16'h0007
MPAM_S	16'h0008
MPAM_NS	16'h0009
MTSX	16'h0010
RN-I	16'h000A
RN-D	16'h000D
RN SAM	16'h000F
HN-P	16'h0011
CXRA	16'h0100
CXHA	16'h0101
CXLA	16'h0102
CCRA	16'h0103
CCHA	16'h0104
CCLA	16'h0105
CCLA_RNI	16'h0106

2.5.3 Child pointers

There is one child pointer register per child node.

The address of the register containing the first child pointer is computed as:

Base node address (of the current 64KB block) + the child_ptr_offset value (from the child_info register).

Each subsequent child pointer register is 8 bytes higher. For more information, see [Figure 2-80 Node structure detail](#) on page 2-166.

For example:

- Base node address = 0x40000.
- Child_ptr_offset in child info register = 0x100.
- Address of first child pointer register (child pointer 0) = Base node address + child_ptr_offset = 0x40100.
- Address to child pointer 1 = Address of child pointer 0 (0x40100) + 0x8 = 0x40108.

Child pointers are 32 bits or less and are contained in the low register. The high register is zero. Child pointer contents include the following:

- The child node address offset from PERIPHBASE (bits 0-29) which is an unsigned integer (positive offset).
- One reserved bit (bit[30]).
- An External Child Node indicator (bit[31]).

For example, address to 64KB block of the child node = PERIPHBASE + child pointer register [29:0]. The child pointer register holds the child node address offset relative to PERIPHBASE.

The External Child Node bit of the child pointer register (bit 31) has the following encodings:

- 1 Indicates that this CHILD POINTER is pointing to a Config Node that is external to CMN-700.
- 0 Indicates that this CHILD POINTER is pointing to a Config Node that is internal to CMN-700.

For CMN-700, external child nodes are only used for CXLA Config Node. The software performing the discovery can use two pieces of information:

1. The CHI node ID corresponding to the Config child node in question.
2. Information in the device port connection information register for the device port that the child node is connected to:
 - a. por_mxp_device_port_connect_info_p0
 - b. por_mxp_device_port_connect_info_p1

————— Note —————

By default, CMN-700 supports two device ports per MXP, P0 and P1. However, you can also extend the number of device ports. For more information, see [Support for extra device ports on MXPs on page 2-69](#).

The device type corresponding to that child node helps the discovery software determine if the child node is RN-F, RN SAM, or CXLA. Every CXRH, CXHA, or CXRA node has a corresponding external CXLA node. Therefore, if the device type is CXRH, CXHA, or CXRA, then the external child node is CXLA. It is the responsibility of the discovery software to ensure that the external child node is powered ON before sending any config accesses to it.

Depending on the size of the mesh (X and Y dimensions), CMN-700 supports three different widths for encoding the X and Y dimension. The number of bits needed is selected based on the larger of the X and Y values.

Table 2-56 Mesh size and encoding bits

Mesh width in X dimension	Mesh width in Y dimension	Number of bits used to encode X, Y
X ≤ 4	Y ≤ 4	2 bits for X, 2 bits for Y
4 < X ≤ 8	Y ≤ 8	3 bits for X, 3 bits for Y
X ≤ 8	4 < Y ≤ 8	3 bits for X, 3 bits for Y
8 < X ≤ 10	8 < Y ≤ 10	4 bits for X, 4 bits for Y

Depending on the size of the mesh (X and Y dimensions), CMN-700 supports two different widths for encoding the X and Y dimension. The number of bits needed is selected based on the larger of the X and Y values.

Table 2-57 Mesh size and encoding bits

Mesh width in X dimension	Mesh width in Y dimension	Number of bits used to encode X, Y
$X \leq 4$	$Y \leq 4$	2 bits for X, 2 bits for Y
$4 < X \leq 8$	$Y \leq 8$	3 bits for X, 3 bits for Y
$X \leq 8$	$4 < Y \leq 8$	3 bits for X, 3 bits for Y

2.5.4 Discovery tree structure

The one-time discovery process creates a lookup table that contains the addresses for all CMN-700 configured devices.

The discovery tree structure consists of three levels:

- Level 1** Root Node, or the HN-D containing the Global Configuration Slave.
- Level 2** XP layer.
- Level 3** Leaf layer with one or two devices.

The following figure shows a 2×2 mesh configuration example with highlighted discovery tree levels.

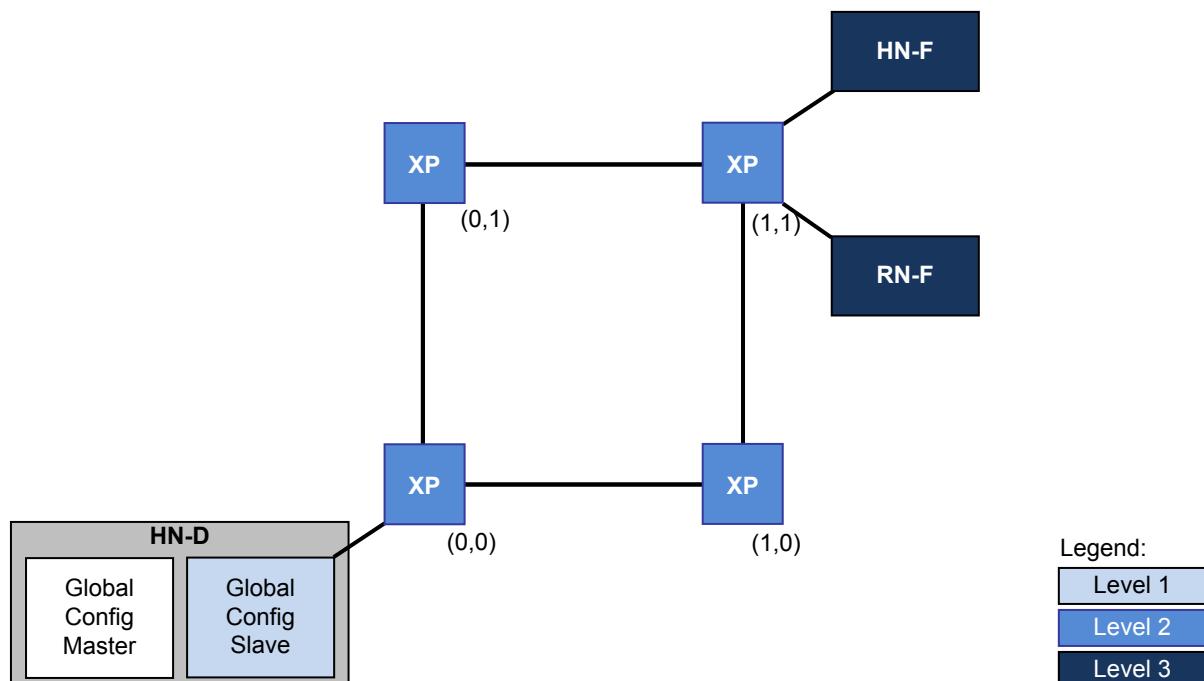


Figure 2-81 2×2 discovery tree example

The following figure shows the discovery tree structure for this 2×2 mesh configuration.

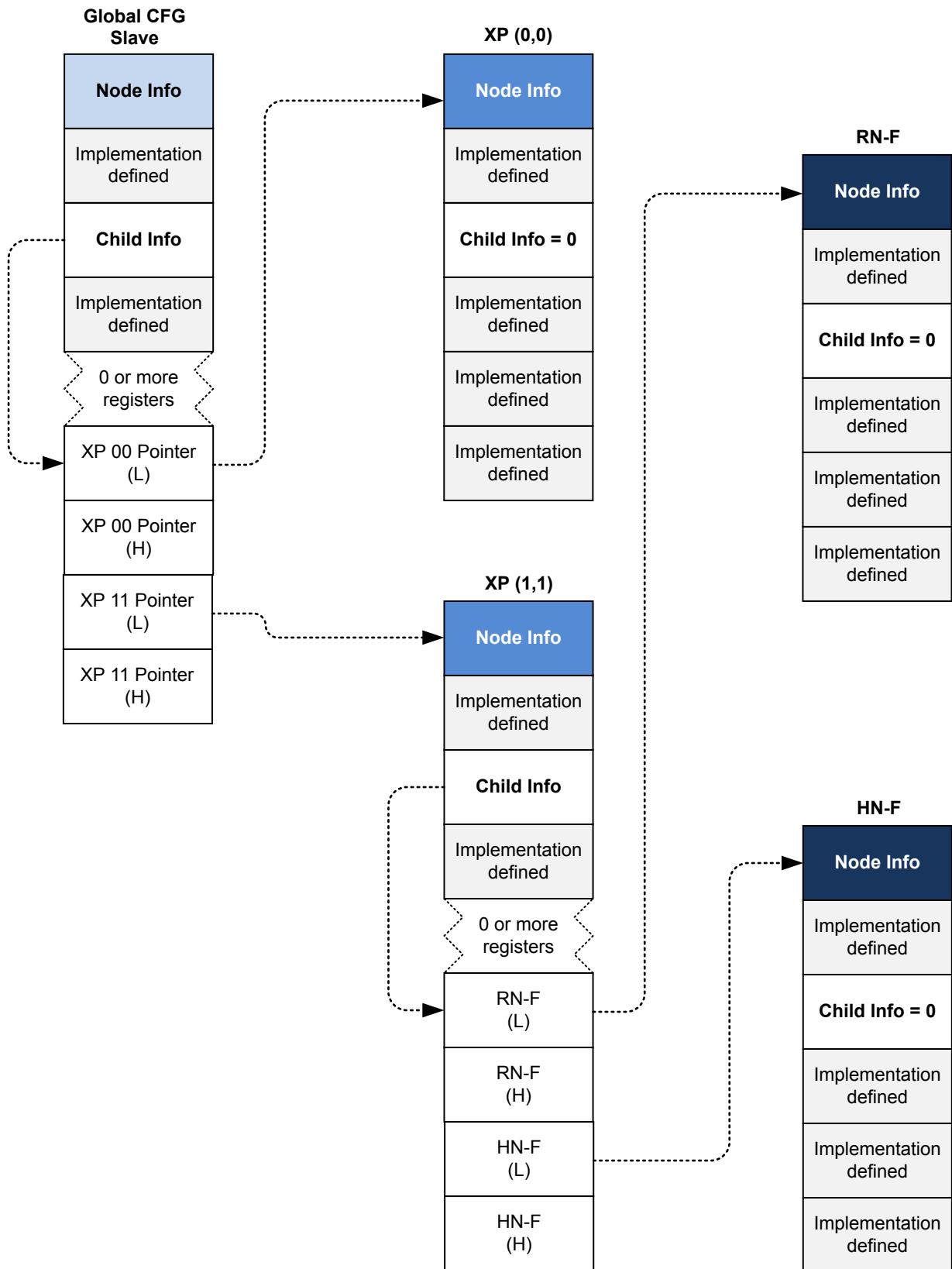


Figure 2-82 2 × 2 discovery tree structure

2.6 Link layer

CMN-700 provides link initialization, flow-control, and link deactivation functionality at the RN-F and SN-F device interfaces.

This functionality comprises the following mechanisms:

- A link initialization mechanism by which the receiving device communicates link layer credits, on each CHI channel that is present, to a transmitting device.
- A flow-control mechanism by which the transmitting device uses link layer credits to send CHI flits – one credit per flit. In turn, the receiving device sends these credits back to the transmitting device, one at a time, when it is done processing each flit to allow for subsequent flit transfers.

Note

The latency (in clock cycles) measured from the time a transmitting device uses a link layer credit to send a flit to the receiving device and the earliest time when it can receive that credit back from the receiving device and send a subsequent flit is called credit roundtrip latency.

- A link deactivation mechanism by which the transmitting device sends all unused link layer credits on each CHI channel back to the receiving device by sending corresponding link flits.

On flit upload channels, RN-F or SN-F is the transmitting device and CMN-700 is the receiving device. On flit download channels, CMN-700 is the transmitting device and RN-F or SN-F is the receiving device.

For a description of the functional requirements of the CHI link layer, see the *AMBA® 5 CHI Architecture Specification*.

This section contains the following subsections:

- [2.6.1 Flit buffer sizing requirements on page 2-171](#).
- [2.6.2 Flit uploads from RN-F or SN-F on page 2-171](#).
- [2.6.3 Flit downloads with RN-F or SN-F on page 2-172](#).

2.6.1 Flit buffer sizing requirements

There are specific size requirements for the CMN-700 flit buffers.

Flit buffer sizing at a receiving device is based on the following two factors:

1. A transmitting device must be able to send flits continuously in a pipelined fashion without stalling due to insufficient link layer credits from the receiving device. This requirement ensures that the system can achieve the full link bandwidth. For a specific system, there is a minimum number of link layer credits that are required so that pipeline stalls can be prevented. You can use the credit roundtrip latency between the transmitting device and receiving device as a measure of the required number of link layer credits.
2. A receiving device must be able to accept and process as many flits as the number of link layer credits it has outstanding at the transmitting device. Therefore, the number of link layer credits that a receiving device sends must not exceed its flit buffering and processing capabilities.

Therefore, flit buffer sizing and corresponding link layer crediting must reflect the credit roundtrip latency. If this requirement is met, the system can achieve optimal flit transfer bandwidth between transmitting and receiving devices. For more information about flit buffer sizing and link layer crediting for flit uploads and downloads at RN-F and SN-F interfaces, see the following sections:

- [2.6.2 Flit uploads from RN-F or SN-F on page 2-171](#)
- [2.6.3 Flit downloads with RN-F or SN-F on page 2-172](#)

2.6.2 Flit uploads from RN-F or SN-F

For flit uploads, the RXBUF_NUM_ENTRIES parameter specifies the number of flit buffers in CMN-700.

For more information about RXBUF_NUM_ENTRIES, refer to [1.5 Configurable options on page 1-19](#).

For optimal flit transfer bandwidth, this parameter must be set equal to the upload credit roundtrip latency ($UpCrdLat<ch>$) which is computed using the following equation.

$UpCrdLat<ch> = UpCrdLatInt<ch> + UpCrdLatExt<ch>$, where:

- $<ch>$ is the CHI channel (REQ, RSP, SNP, or DAT).
- $UpCrdLatInt<ch>$ is the upload credit latency inside CMN-700. This latency is measured (in clock cycles) from the time $RX<ch>FLITV$ input is asserted by the RN-F or SN-F for a flit uploaded to CMN-700 to the earliest time when $RX<ch>LCRDV$ output is asserted by CMN-700 to the RN-F or SN-F after the flit is processed and the credit sent back. At the RN-F/SN-F interfaces, $UpCrdLatInt<ch> = 1$ on all CHI channels.
- $UpCrdLatExt<ch>$ is the upload credit latency outside CMN-700. This latency is measured (in clock cycles) from the time $RX<ch>LCRDV$ output is asserted by CMN-700 when the credit is sent back to the RN-F or SN-F to the earliest time when $RX<ch>FLITV$ input is asserted by the RN-F or SN-F when the credit is used to send a subsequent flit.

2.6.3 Flit downloads with RN-F or SN-F

For optimal flit downloads, the RN-F or SN-F must size its input buffers to reflect the download credit roundtrip latency ($DnCrdLat<ch>$).

$DnCrdLat<ch>$ is computed using the following equation.

$DnCrdLat<ch> = DnCrdLatInt<ch> + DnCrdLatExt<ch>$, where:

- $<ch>$ is the CHI channel (REQ, RSP, SNP, or DAT).
- $DnCrdLatInt<ch>$ is the download credit latency inside CMN-700. This latency is measured (in clock cycles) from the time $RX<ch>LCRDV$ input is asserted by the RN-F or SN-F to CMN-700 to the earliest time when $RX<ch>FLITV$ output is asserted by CMN-700 to the RN-F or SN-F for a flit using that credit. At the RN-F or SN-F interfaces, $DnCrdLatInt<ch> = 2$ on all CHI channels.
- $DnCrdLatExt<ch>$ is the download credit latency outside CMN-700. This latency is measured (in clock cycles) from the time $RX<ch>FLITV$ output is asserted by CMN-700 for a flit downloaded to the RN-F or SN-F to the earliest time when $RX<ch>LCRDV$ input is asserted when the corresponding credit is returned by the RN-F or SN-F to CMN-700.

Note

When MPU is enabled, an extra cycle of latency is present in the SNP flit path to RN-Fs.

2.7 PCIe integration

CMN-700 supports integration of a PCIe *Root Complex* (RC) or *EndPoint* (EP).

This section contains the following subsections:

- [2.7.1 PCIe topology requirements on page 2-173](#).
- [2.7.2 PCIe master and slave restrictions and requirements on page 2-173](#).
- [2.7.3 System requirements for PCIe devices on page 2-174](#).

2.7.1 PCIe topology requirements

There are specific topology rules that you must follow when integrating PCIe components with CMN-700.

The following PCIe topology requirements apply:

- PCIe slaves must not be connected to HN-D.
- PCIe slaves must not share HN-I with other non-PCIe slaves.
- HN-P must only be used to connect to PCIe slaves.

2.7.2 PCIe master and slave restrictions and requirements

There are restrictions on both the types and flow of transactions between PCIe devices and CMN-700.

————— Note ————

In this section, PCIe HN-I refers to an HN-I or HN-P which has a PCIe slave that is connected to it. HN-I refers to all other HN-Is.

CMN-700 supports peer-to-peer PCIe traffic. This function allows one PCIe endpoint to communicate with another PCIe endpoint through the interconnect.

Transaction type restrictions

A PCIe master must not send any *Non-Posted Configuration and I/O Writes* (NPR-Wr) targeting CMN-700.

Flow control requirements from CMN-700 to PCIe slave

The PCIe slave must be able to sink at least one NPR-Wr from CMN-700 sent on the PCIe HN-I AXI/ACE-Lite master port. This requirement guarantees that the PCIe HN-I AW channel remains unblocked. Therefore, *Posted Writes* (P-Wrs) from PCIe master targeting the downstream slave device can progress, as required by the PCIe ordering rules.

Flow control requirements from PCIe master to CMN-700

Your configuration might have a *System Memory Management Unit* (SMMU) or GIC-ITS in the path between the PCIe master interface and the RN-I slave interface. If using this configuration, *Non-Posted Reads* (NPR-Rds) from the PCIe master must not target any PCIe HN-I. You can also use one of the following mutually exclusive flow control options:

- Use a separate master interface port in the SMMU and GIC-ITS for translation table walks (TCU in MMU-600 or GIC-600 and beyond). You can then connect this port to a different RN-I which does not send any requests to any PCIe HN-I. None of the masters that are connected to the RN-I can talk to any PCIe HN-I.
- When per port reservation is enabled use multiple AXI ports within the same RN-I or RN-D to connect SMMU or GIC-ITS for translation table walk. Per port reservation is described in the following section.

Enable per port reservation by clearing the `dis_port_token` of the `por_{rni,rnd}_cfg_ctl` register to 0. This programming enables reservation for all ports and also for read and write channels.

When per port reservation is enabled, each port has at least one reserved entry in both read and write transaction trackers. The per port reservation guarantees progress of requests through each port. You can increase the number of reserved entries according to your bandwidth needs by programming the following register fields:

- s<X>_rd_token field of the por_{rni,rnd}_s<X>_port_control register.
- s<X>_wr_token field of the por_{rni,rnd}_s<X>_port_control register.

For more information, see [por_rnd_s_0-2_port_control](#) on page 3-260 and [por_rni_s_0-2_port_control](#) on page 3-953.

2.7.3 System requirements for PCIe devices

There are certain system-level requirements that you must meet when integrating PCIe devices with CMN-700. These requirements determine which CMN-700 devices can handle certain request types and how PCIe and non-PCIe transactions must be handled.

————— Note ————

Program the field s{x}_tablewalk_mstr_present in config register por_{rni,rnd}_s{x}_port_control (x=0,1,2) for the corresponding port where the SMMU/GIC table walker is connected.

————— Note ————

In this section, PCIe HN-I refers to an HN-I or HN-P which has a PCIe slave connected to it. HN-I refers to all other HN-Is.

CMN-700 has the following system requirements for PCIe devices:

- All non-PCIe I/O slave devices must complete all writes without creating any dependency on a transaction in the PCIe subsystem.
- Your configuration might have an SMMU in the path between the PCIe master interface and the RN-I slave interface. If using this kind of configuration, table-walk requests from the SMMU can only be sent to memory through the HN-F or non-PCIe HN-I.
- Interrupt translation table walk requests from GIC-ITS can only be sent to memory through the HN-F or non-PCIe HN-I.

There are certain programming requirements that your system must meet to ensure proper PCIe functionality. For more information, see [3.4.6 RN-I and HN-I PCIe programming sequence](#) on page 3-1244.

2.8 Reliability, Availability, and Serviceability

The CMN-700 *Reliability, Availability, and Serviceability* (RAS) features are implemented as set of distributed logging and reporting registers and a central interrupt handling unit.

The distributed logging and reporting registers are associated with devices that can detect errors. These devices are XP, HN-I, HN-F, SBSX, CCHA, and CXHA.

The central interrupt handling unit is located in the HN-D.

Each device that can detect errors logs the errors in local registers. The device sends error information to the central interrupt handling unit in the HN-D. The HN-D contains four sets of five error groups, which are based on the device type of the error source. The sets consist of a Secure and Non-secure group for errors, and a Secure and Non-secure group for fault-type errors. The groups are represented by *ERRor Group Status Registers* (ERRGSRs).

Each device type has up to 16 ERRGSRs, depending on how many devices of that type are present in the CMN-700 system. For example, the following table shows a possible configuration of the MXP ERRGSRs.

Table 2-58 Example MXP ERRGSR configuration

ERRGSR name	Register offset	Error group
por_cfgm_errgsr_mxp_0	16'h3000	MXP_<63:0> error status
por_cfgm_errgsr_mxp_1	16'h3008	MXP_<63:0> fault status
por_cfgm_errgsr_mxp_2	16'h3010	MXP_<127:64> error status
por_cfgm_errgsr_mxp_3	16'h3018	MXP_<127:64> fault status
por_cfgm_errgsr_mxp_0_NS	16'h3040	MXP_<63:0> error status NS
por_cfgm_errgsr_mxp_1_NS	16'h3048	MXP_<63:0> fault status NS
por_cfgm_errgsr_mxp_2_NS	16'h3050	MXP_<127:64> error status NS
por_cfgm_errgsr_mxp_3_NS	16'h3058	MXP_<127:64> fault status NS

If CMN-700 has ≤ 64 MXPs, only por_cfgm_errgsr_mxp_0, por_cfgm_errgsr_mxp_1, por_cfgm_errgsr_mxp_0_NS, and por_cfgm_errgsr_mxp_1_NS are present.

The following figure shows the five error groups, and the four respective interrupt request signals, with XP connections highlighted. The HN-I, HN-F, SBSX, CCHA, and CXHA use the same input/output structure.

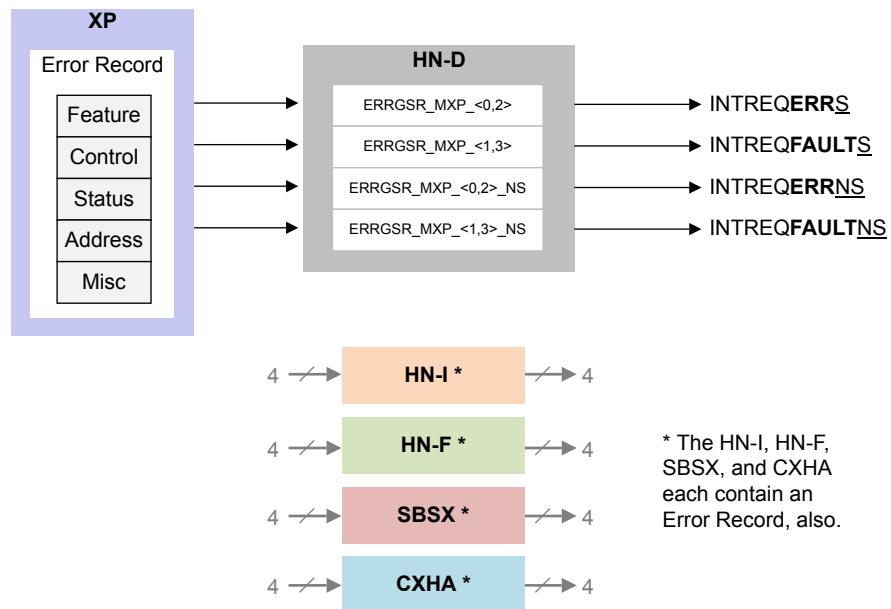


Figure 2-83 Error top-level diagram

Each device that can detect errors has five Error Record registers that contain the error type, along with other information such as the address and opcode. Error types include *Corrected Error* (CE), *Deferred Error* (DE), and *Uncorrected Error* (UE).

For more information on error types, see [2.8.1 Error types](#) on page 2-177.

For register details, see [3.3 Register descriptions](#) on page 3-249.

Error interrupt handler flow example

The following sequence of events and figure describe the process for determining the error source and type of an HN-I generating an interrupt request:

1. The HN-D generates an interrupt for one of the five error group types.
2. The error group indicates the error source device type, which can be:
 - XP
 - HN-I, which is used in this case
 - HN-F
 - SBSX
 - CCHA
 - CXHA
3. The bit location within the error group indicates the logical ID of that device type. In this case, it reveals an HN-I error, the HN-I Error Record Status block for this example.
4. The Status block of the Error Record for the specific XP, HN-I, HN-F, SBSX, or CXHA indicates the type of error.
5. The Address and Misc blocks of the Error Record provide further details regarding error root cause, in this case a Corrected Error.
6. The Valid bit is also asserted.
7. To clear the asserted interrupt on the pin, the valid bit of the error status has to be cleared.

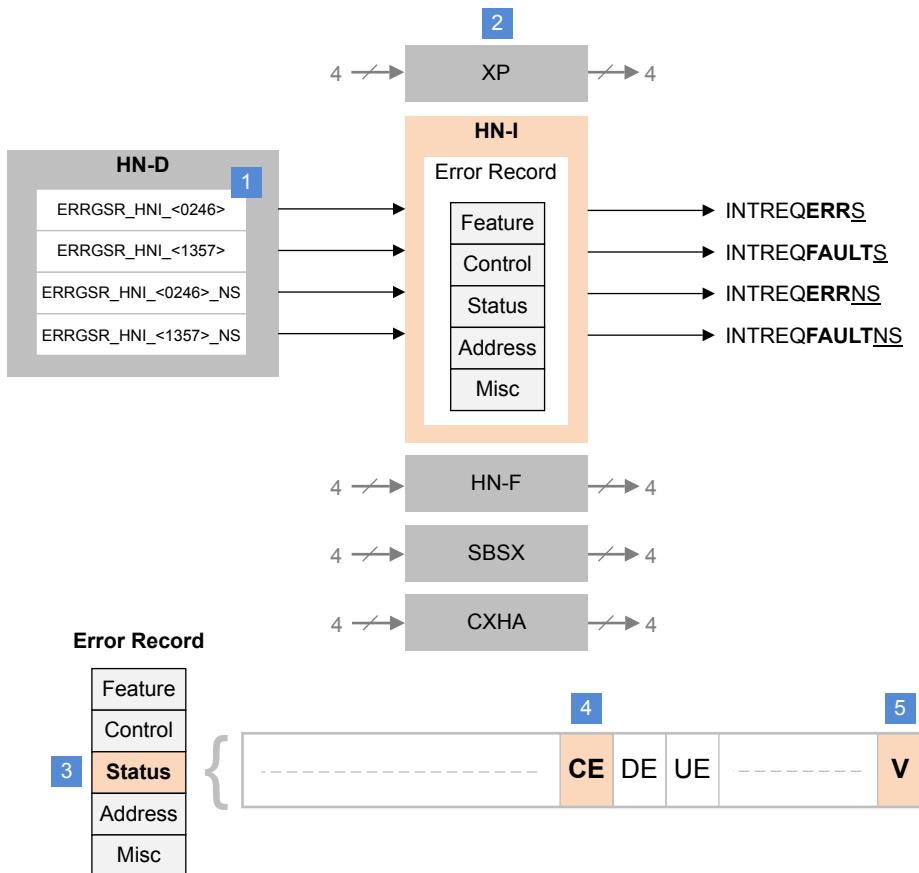


Figure 2-84 Error interrupt handler flow example

This section contains the following subsections:

- [2.8.1 Error types on page 2-177](#).
- [2.8.2 Error Detection and Deferred Error values on page 2-179](#).
- [2.8.3 Error detection, signaling, and reporting on page 2-180](#).
- [2.8.4 Error reporting rules on page 2-184](#).
- [2.8.5 HN-F error handling on page 2-184](#).
- [2.8.6 HN-I error handling on page 2-185](#).
- [2.8.7 SBSX error handling on page 2-188](#).
- [2.8.8 MTU error handling on page 2-188](#).
- [2.8.9 RN-I error handling on page 2-190](#).
- [2.8.10 XP error handling on page 2-190](#).
- [2.8.11 CXHA error handling on page 2-192](#).
- [2.8.12 CCIX Protocol Error messaging support on page 2-193](#).

2.8.1 Error types

CMN-700 supports several error types.

The supported errors are:

- *Corrected Error* (CE).
- *Deferred Error* (DE).
- *Uncorrected Error* (UE).

— Note —

CEs, DEs, and UEs can occur simultaneously.

There might be cases when an error occurs and sets the status register, however, the interrupt reporting is not enabled. For other cases where the interrupt asserts, the interrupt request is generated immediately when enabled. Otherwise, if interrupt reporting is disabled, any interrupt is cleared and the error remains logged with UE, DE, and CE.

— Note —

If both ERRCTLR.UI (uncorrected interrupt) and ERRCTLR.FI (fault interrupt) are set and a UE occurs, both fault and error interrupts are delivered from CMN-700.

Correctable Errors

These errors can be corrected using *Error Correcting Code* (ECC) or other methods. They include:

- A single-bit ECC error.
- An error that is recovered by replaying the transaction in the pipeline.

The system handles these errors as follows:

1. Detects the error and increments the ERMISC.CEC counter. Sets ERRSTATUS.AV and ERRSTATUS.MV. Logs the attributes in the ERRADDR and ERMISC registers.
2. If CEC counter overflowed, the system updates ERRSTATUS.V and ERRSTATUS.CE, and sets ERMISC.CECOF.
3. Masks signaling of the error to the *RAS Control Block* (RCB) using ERRCTLR.CFI.
4. If there are multiple CEC overflows, then the system sets ERRSTATUS.OF.

Correctable Errors

Single-bit *Error Correcting Code* (ECC) errors can be corrected using ECC or other methods. The system handles these errors by completing the following steps:

1. Detects the error and increments the ERMISC.CEC counter. Sets ERRSTATUS.AV and ERRSTATUS.MV. Logs the attributes in the ERRADDR and ERMISC registers.
2. If CEC counter overflowed, the system updates ERRSTATUS.V and ERRSTATUS.CE, and sets ERMISC.CECOF.
3. Masks signaling of the error to the *RAS Control Block* (RCB) using ERRCTLR.CFI.
4. If there are multiple CEC overflows, then the system sets ERRSTATUS.OF.

Deferred Errors

These errors are UEs that have the following properties:

- Detected in one node of CMN-700, but the data is not used within the same node.
- Poison bits are set for the data.

The errors can be either fatal or non-fatal errors as described in this section. These errors are not correctable, but the system can operate for a time without being corrupted. These errors can be contained and the system might be able to recover using software means. They include:

- A data double bit ECC error in the SLC Data RAM.
- Data check error detected in SLC.

The system handles these errors as follows:

1. Logs the error information in the applicable ERRSTATUS, ERRADDR, and ERMISC registers.
2. Sets ERRSTATUS.V and ERRSTATUS.DE.
3. Masks signaling of the error to the RAS control block using ERRCTLR.FI and ERRCTLR.UI.
4. If there are multiple DEs, then the system sets ERRSTATUS.OF.

Uncorrectable Fatal Errors

These errors are in the control logic at a node. Continuing operation might corrupt the system beyond recovery. They include:

- A double-bit ECC error in SLC tag.
- Flit parity error.
- *Non-Data Error* (NDE) in a response packet.

The system handles these errors as follows:

1. Logs the error information in the applicable ERRSTATUS, ERRADDR, and ERMISC registers.
2. Sets ERRSTATUS.V and ERRSTATUS.UE.
3. Masks signaling of the error to the RAS control block using ERRCTLR.UI.
4. If there are multiple UEs, then the system sets ERRSTATUS.OF.

A component might not respond to further messages after an error is signaled. In this case, for the error handling routine to be successful, the component must still respond to configuration access requests from the configuration bus.

Note

When a subsequent error updates ERRSTATUS register, if the previous error has AV set, but the new error does not have address associated with it, the AV is not cleared.

CMN-700 follows the *Arm® Reliability, Availability, and Serviceability (RAS) Specification Armv8, for the Armv8-A architecture profile* for mapping of the different error types to the interrupt. The following table summarizes the mapping of various error types.

Table 2-59 Mapping of error types

Interrupt type	Error type		
	Uncorrected Error	Detected Error	Corrected Error
Fault Handling Interrupt	Yes (if ERRCTLR.FI==1)	Yes (if ERRCTLR.FI==1)	Yes (if ERRCTLR.CFI==1)
Error Recovery Interrupt	Yes (if ERRCTLR.UI==1)	No	No

2.8.2 Error Detection and Deferred Error values

For XP, the default values of *Error Detection* (ED) and DE depend on build time parameters.

Only the HN-F has CE counters that are implemented in the ERMISC register. The default values of UI, FI, and CFI are 2'b10, which enables control for the interrupt generation. The following table contains default values of ED and DE for XP.

Table 2-60 Default values of ED and DE for XP

POR_FLIT_PAR_EN	POR_DATACHECK_EN	MXP_DEV_DATACHECK_EN		ED	DE
		P0	P1		
0	0	0	0	2'b00	2'b00
0	1	0	0	2'b01	2'b01
		0	1	2'b01	2'b01
		1	0	2'b01	2'b01
		1	1	2'b00	2'b00
1	0	0	0	2'b01	2'b00

Table 2-60 Default values of ED and DE for XP (continued)

POR_FLIT_PAR_EN	POR_DATACHECK_EN	MXP_DEV_DATACHECK_EN		ED	DE
		P0	P1		
1	1	0	0	2'b01	2'b01
		0	1	2'b01	2'b01
		1	0	2'b01	2'b01
		1	1	2'b01	2'b00

Table 2-61 Default values of ED and DE for XP

POR_FLIT_PAR_EN	POR_DATACHECK_EN	MXP_DEV_DATACHECK_EN		ED	DE
		P0	P1		
0	0	0	0	2'b00	2'b00

For SBSX, if the AXDATAPOISON_EN parameter is not set, the default values of ED and DE are 2'b01. If the parameter is set, the default values are 2'b00.

For HN-I and HN-F, the default values of ED and DE are always 2'b01.

All fields are required, even though only HN-F has CE counters that are implemented in ERRMISC.

The default values of UI, FI, and CFI must be 2'b10, indicating that the interrupt generation is controllable.

2.8.3 Error detection, signaling, and reporting

Each CMN-700 component that is connected to a configuration bus can be included in the local error reporting mechanism.

The error handling protocol is as follows:

- Error overflow
- ERRSTATUS.OF value after errors
- ERRMISC fields and register bits

Error overflow

The overflow is set when different types of errors are detected. It is also asserted when multiple errors of equal priority are detected.

0b1 More than one error has been detected.

0b0 Only one error of the most significant type that ERRSTATUS.{UE, CE, DE} describes has been detected.

This bit is read/write-one-to-clear.

————— Note —————

ERRSTATUS.OF is only for the highest priority error. For example, if another DE follows the first DE, ERRSTATUS.OF is set. When the next UE happens, ERRSTATUS.OF is cleared. ERRSTATUS.OF is cleared because UE is the highest priority error in the system, and is the first occurrence of UE.

The following table shows the value of ERRSTATUS.OF after errors occur at t0, t1, and t2.

Table 2-62 ERRSTATUS.OF value after errors

Error at			Status of OF after error		
t0	t1	t2	t0	t1	t2
CE	CE	CE	0	1	1
CE	CE	DE	0	1	0
CE	CE	UE	0	1	0
CE	DE	CE	0	0	0
CE	DE	DE	0	0	1
CE	DE	UE	0	0	0
CE	UE	CE	0	0	0
CE	UE	DE	0	0	0
CE	UE	UE	0	0	1
DE	CE	CE	0	0	0
DE	CE	DE	0	0	1
DE	CE	UE	0	0	0
DE	DE	CE	0	1	1
DE	DE	DE	0	1	1
DE	DE	UE	0	1	0
DE	UE	CE	0	0	0
DE	UE	DE	0	0	0
DE	UE	UE	0	0	1
UE	CE	CE	0	0	0
UE	CE	DE	0	0	0
UE	CE	UE	0	0	1
UE	DE	CE	0	0	0
UE	DE	DE	0	0	0
UE	DE	UE	0	0	1
UE	UE	CE	0	1	1
UE	UE	DE	0	1	1
UE	UE	UE	0	1	1

ERRMISC fields

ERRMISC is the Secondary Error Syndrome Register. The fields of this register differ for ECC, parity, and other errors. The following table summarizes the valid fields for each unit.

Table 2-63 ERRMISC register bits

Bit	Component				
	XP	HN-I	HN-F	SBSX	CXHA
63	-	-	CECOF	-	-
62			SETMATCH		
61			-		
60			ERRSET[12:0]		
59					
58	TGTID[10:0]				
57					
56		LPID[4:0]			
55					
54					
53					
52					
51		-			
50					
49		ORDER[1:0]			
48					
47	-	-	CEC[15:0]	-	
46					
45					
44					
43					
42					
41					
40					
39					
38					
37					
36					
35					
34					
33					
32					

Table 2-63 ERRMISC register bits (continued)

Bit	Component				
	XP	HN-I	HN-F	SBSX	CXHA
31	-	-	-	-	-
30		SIZE[2:0]		SIZE[2:0]	
29					
28					
27		MEMATTR[3:0]		MEMATTR[3:0]	
26					
25					
24					
23		-		-	
22					
21	OPCODE[5:0]	OPCODE[5:0]			
20					
19					
18					
17			OPTYPE[1:0]		
16				OPTYPE	
15	-	-	-	-	
14	SRCID[10:0]	SRCID[10:0]	SRCID[10:0]	SRCID[10:0]	
13					
12					
11					
10					
9					
8					
7					
6					
5					
4					
3	-	ERRSRC[3:0]	ERRSRC[3:0]	-	-
2	ERRSRC[2:0]				
1					ERRSRC[1:0]
0					

Error log clearing

In addition to the Error Syndrome Registers, each component has a write-only Error Syndrome Clear Register. Write the applicable mask bits to clear the first_err_vld and mult_err bits of the Error Syndrome 0 Register.

2.8.4 Error reporting rules

CMN-700 uses specific error reporting rules, concerning which errors must be reported and propagated.

The rules regarding error reporting in CMN-700 are:

- Any error originating in CMN-700 is reported.
- Any error originating outside CMN-700 but corrupting CMN-700 is reported.
- The HN-I can report an error in a response packet from outside CMN-700 if it does not propagate the response any further.
- All non-posted write errors are propagated where possible.

2.8.5 HN-F error handling

Errors are reported at the HN-F for various reasons.

Request errors at HN-F

The HN-F detects:

- ECC errors in SF Tag, SLC Tag, and Data RAMs.
- Data check and poison errors on DAT flits.
- *Non-Data Errors* (NDEs) on responses.

ECC errors in SF Tag, SLC Tag, and Data RAMs

HN-F detects single-bit and double-bit ECC errors in the SF Tag, SLC Tag, and Data RAMs. It can correct single-bit ECC errors. Such errors are logged and reported as CEs.

The source of the double-bit ECC errors determines how they are handled.

SLC Data RAM

- Logged and reported as DEs.
- Propagated to the data consumer in the form of data poison.

SF Tag RAM

- Logged and reported as DEs.
- Not propagated to the requestor.
- The SF Tag RAM in the HN-F is disabled following the first occurrence of the double-bit ECC error.

SLC Tag RAM

- Fatal error.
- Logged and reported as UEs.
- Propagated to the requestor as NDEs in the responses.

Data check and poison errors on DAT flits

If data is allocated by the HN-F, the HN-F detects data check errors and poison error on the data flits. In such cases, HN-F logs and reports the data check error as a DE. If HN-F allocated the data in SLC Data RAM, it converts the data check error into data poison for all subsequent requests to this cache line.

If por_hnf_aux_ctl.hnf_poison_intr_en == 1, then the poison errors originating from HN-F are logged and reported as UEs.

Poison errors on DAT flits

If data is allocated by the HN-F, the HN-F detects poison errors on the data flits.

If por_hnf_aux_ctl.hnf_poison_intr_en == 1, then the poison errors originating from HN-F are logged and reported as UEs.

NDEs on responses

HN-F can receive NDEs from other data and response sources such as RN-F, RN-I, and SN-F. If the cache line was allocated in SLC Data RAM, it is logged and reported as a UE. If the cache line is not allocated in HN-F SLC, it propagates the errors to the requestor as an NDE.

2.8.6 HN-I error handling

Errors are reported at the HN-I for various reasons.

Request errors at HN-I

The HN-I detects errors on receiving various request types and sends an NDE response to the requesting RN.

The HN-I logs request information in the error logging registers, por_hni_erraddr(_NS) and por_hni_errmisc(_NS). They are marked as DEs in the error status register, por_hni_errstatus(_NS). The HN-I detects errors on receiving the following request types:

- Coherent Read.
- CleanUnique/MakeUnique.
- Coherent/CopyBack Write.
- Atomic.
- Illegal Configuration Read/Write (HN-D only).
- Unsupported Exclusive access (HN-P only).

The reqerr_cohreq_en configuration bit in the por_hni_cfg_ctl register enables or disables the sending of NDE responses and logging of error information for following request types. By default, this bit is enabled. It can only be programmed during boot time to any one of the following:

- Coherent Read.
- CleanUnique/MakeUnique.
- Coherent/CopyBack Write.

The HN-P has a configuration bit, the disable_hnp_excl_err bit, in the por_hni_cfg_ctl register. This bit disables the sending of NDE responses and logging of error information for unsupported Exclusive accesses. Exclusive WriteNoSnp and Exclusive ReadNoSnp requests are unsupported in HN-P as HN-P is not a PoS device. Disabling this error results in an Exclusive Pass response to these requests.

The following table lists all the requests that an HN-I detects as errors and the support of reqerr_cohreq_en.

Table 2-64 HN-I request errors and support for configuration bit

Request type	Reqerr_cohreq_en controls sending of NDE and log error
Coherent Read	Yes
CleanUnique/MakeUnique	Yes
Coherent/CopyBack Write	Yes
Atomics	No
Illegal Configuration Read/Write	No

- Coherent reads are downgraded to ReadNoSnp and sent downstream (AXI/ACE-Lite slave).
- Coherent/Copyback writes are downgraded to WriteNoSnp and sent downstream (AXI/ACE-Lite slave).
- Illegal Configuration Read is sent as ReadNoSnp to downstream (AXI/ACE-Lite slave).

- CleanUnique, MakeUnique, Atomics, and Illegal Configuration Writes are handled within HN-I.
- StashOnceShared, StashOnceUnique, and PrefetchTgt are completed within HN-I without any errors.

Data Errors at HN-I

The HN-I only detects errors on write data if it does not detect an error on that request.

The following provides an overview of AXI and ACE-Lite write requests and configuration write requests, with no request error:

- For AXI and ACE-Lite write requests with no request error, when they receive Poison error on data, the HN-I detects the error. If downstream does not support poison, the HN-I logs the request information in por_hni_erraddr(_NS) and por_hni_errmisc(_NS). The write requests are marked as UEs in the error status register, por_hni_errstatus(_NS).
- For configuration write requests with no request error, on receiving write data with Partial ByteEnable error, Data check error, or Poison, HN-I detects and sends an NDE response to the requesting RN. It logs the SrcID and TxnID of the request and drops the write. They are marked as DEs in the error status register, por_hni_errstatus(_NS).

————— Note ————

StashOnceShared, StashOnceUnique, and PrefetchTgt are completed within HN-I without any errors.

Response Errors at HN-I

The HN-I only detects errors on write response if it does not detect an error on that request.

To summarize:

- For AXI/ACE-Lite write requests with early completions from HN-I and no request error, on receiving *Slave Error* (SLVERR), or *Decode Error* (DECERR) on downstream write response (BRESP), HN-I detects the error. It logs request information in por_hni_erraddr(_NS) and por_hni_errmisc(_NS). They are marked as UEs in the error status register, por_hni_errstatus(_NS).
- For AXI/ACE-Lite write requests with downstream completions and no request error, SLVERR, or DECERR on downstream write response (BRESP) are passed on to the requesting RN as CHI DEs or NDEs.
- For AXI/ACE-Lite read requests, SLVERR and Poison (if supported by downstream) are both converted to Poison within the CMN-700 system, independent of error on request. DECERRs on downstream read responses are passed on to the requesting RN.

HN-I summary on sending NDE and DE

The HN-I sends NDE scenarios for certain situations.

The HN-I sends NDE in the following cases:

- Request Error.
 - Coherent Read (if reqerr_cohreq_en is set to 1).
 - CleanUnique/MakeUnique (if reqerr_cohreq_en is set to 1).
 - Coherent/CopyBack Write (if reqerr_cohreq_en is set to 1).
 - Atomic.
 - Illegal Configuration Read/Write (HN-D only).
 - Unsupported Exclusive access (HN-P only).

————— Note ————

For the legal format of configuration read/write request, refer to [3.1.5 Requirements of configuration register reads and writes on page 3-216](#).

- Write Data Error for Configuration Write request (HN-D only).

- Partial ByteEnable Error.
- Data Check Error.
- Poison.
- AXI/ACE-Lite Response Error.
 - DECERR on *downstream write response (BRESP)* for writes with downstream completions.
 - DECERR on *downstream read response (RRESP)*.
- Illegal MPU Access Error.
 - Coherent Read.
 - Coherent Write.
 - CleanUnique.
 - MakeUnique.
 - StashOnce*.
 - Atomic.

————— Note ————

The HN-I sends NDE regardless of whether BUS_ERROR was requested.

The HN-I sends DE in the following cases:

- AXI/ACE-Lite Response Error.
 - SLVERR on **BRESP** for writes with downstream completions.

HN-I summary on logging errors

The HN-I logs an error as deferred or uncorrected in certain conditions.

The HN-I logs errors for all illegal MPU accesses detected. To disable this behavior, set por_hni_aux_ctl.disable_mpu_err_logging to 0.

Deferred Errors

The HN-I logs an error as deferred in the following cases:

- Request error.
 - Coherent Read (if reqerr_cohreq_en is set to 1).
 - CleanUnique/MakeUnique (if reqerr_cohreq_en is set to 1).
 - Coherent/CopyBack Write (if reqerr_cohreq_en is set to 1).
 - Atomic.
 - Illegal Configuration Read/Write.
 - Unsupported Exclusive access (HN-P only). This error type is disabled if disable_hnp_excl_err is set to 1.

————— Note ————

For the legal format of a Configuration Read/Write request, refer to [3.1.5 Requirements of configuration register reads and writes on page 3-216](#).

- Write Data Error for Configuration Write request.

- Partial ByteEnable Error.
- Data Check Error.
- Poison Error.

Uncorrected Errors

The HN-I logs an error as uncorrected in the following cases:

- Write Data Error for AXI/ACE-Lite write requests.
 - Poison Error on data if downstream does not support poison.
- AXI/ACE-Lite Write Response Error.
 - SLVERR or DECERR on BRESP for writes that were sent early completions.

CML configuration with HN-I

In CML configuration, HN-I must be configured to report NDE response on coherent requests.

This requirement is met by setting por_hni_cfg_ctl.reqerr_cohreq_en. This action is required in CML mode so that NDE error responses are not missed on CCIX because of early completion responses from the CXG block.

2.8.7 SBSX error handling

This section describes how errors are handled at the SBSX.

If the AXI memory controller downstream of SBSX does not support POISON (indicated by por_sbsx_unit_info.axdata_poison_en = 0), and if CHI Write Data has Poison set, then SBSX detects and logs this error.

If por_sbsx_cfg_ctl.sbsx_rpt_err_on_poison_rd = 1, and if SBSX receives Poison on read data from downstream, then SBSX detects and logs this error.

————— Note —————

SBSX does not have opcode-based Request/Response Error class as does HN-I.

The following table shows the SBSX summary on sending an NDE/DE.

Table 2-65 SBSX summary on sending NDE/DE

Case number	Source of error	SBSX error response
1	Decode Error on RDATA from AXI side	NDE on COMP_DATA on CHIE side
2	Slave Error on RDATA from AXI side	Poison on COMP_DATA on CHIE side
3	Decode Error on BRESP from AXI side	NDE on COMP for CMOs or Writes with EWA=0
4	Slave Error on BRESP from AXI side	DE on COMP for CMOs or Writes with EWA=0

2.8.8 MTU error handling

Errors at MTU are reported for various reasons. All the errors are logged in MTU RAS registers.

————— Note —————

Both the MTU and SBSX error handling functionality are combined to provide the full MTSX error handling functionality. When an MTSX error is reported, both SBSX and MTU RAS registers must be checked to determine the source of the error. Also, when clearing the error, the correct set of RAS registers must be cleared. For more information about SBSX error handling functionality, see [2.8.7 SBSX error handling on page 2-188](#).

When multiple MTU errors are detected in the same cycle, the RAS registers can only store the details of one error. Which error is saved depends on the priority of the error.

The following table shows the priority of the *Uncorrectable Errors* (UEs) that are reported at the MTU:

Table 2-66 Priority-ranking of UEs at the MTU

UE type	Priority
Tag control RAM double-bit error	1 (highest)
Tag data RAM double-bit error	2
AXI-R slave error	3
AXI-R decode error	4
AXI-R poison	5
AXI-R DataCheck	6
AXI-B slave error	7
AXI-B decode error	8
Data PA out of bounds (if enabled)	9 (lowest)

The following table shows the priority of the *Correctable Errors* (CEs) that are reported at the MTU:

Table 2-67 Priority-ranking of CEs at the MTU

UE type	Priority
Tag control RAM single-bit error	1 (highest)
Tag data RAM single-bit error	2 (lowest)

The following table shows the only circumstances where MTU sends an NDE response back to the requestor and how the MTU responds.

Table 2-68 Conditions when MTU returns NDE response to the requestor

Source of error	MTU error response
Tag RAM control double-bit error for read request	NDE on COMP_DATA on CHI-E
Tag RAM control double-bit error for tag match request	NDE on match response on CHI
AXI-R decode error for read request	NDE on COMP_DATA on CHI-E
AXI-R decode error for tag match request	NDE on match response on CHI-E
AXI-R decode error for CMO with late completion	NDE on COMP on CHI-E
AXI-B decode error for CMO with late completion	NDE on COMP on CHI-E

The MTU tag data RAM handles specific error types in the following way:

- Single-bit ECC errors are reported and logged.
- Corrected data after single-bit ECC error is written back into the same way.
- Double-bit ECC errors are reported and logged.
- Data with double-bit ECC is marked invalidated. This behavior could mean that modified tags are lost.

The MTU tag control RAM handles specific error types in the following way:

- Single-bit ECC errors are reported and logged.
- Corrected control bits after single-bit ECC error are written back into same way.
- Double-bit ECC errors are fatal.

- Double-bit ECC errors are reported and logged.
- RAM way with control double-bit ECC is marked invalidated. This behavior could mean that modified tags are lost.

The MTU handles AXI read response (denoted as AXI-R) errors in the following way:

- Slave, poison, and Data Check errors are reported and logged. These errors are not propagated any further.
- Decode errors are reported and logged. MTU propagates the error to the requestor for tag read (transfer), tag Match (match response), and CMO (if late completion).
- If AXI read is required to merge with tag update, modified tags could be lost.

The MTU handles AXI write response (denoted as AXI-B) errors in the following way:

- Slave, Poison, and Data Check errors are reported and logged. These errors are not propagated any further.
- Decode errors are reported and logged. MTU propagates the error to the requestor for late completion CMO.

2.8.9 RN-I error handling

RN-I does not report any errors. When a parity error is detected in the *Read Data Buffer* (RDB) RAMs, RN-I or RN-D propagates the error on AXI R channel as **RPOISON** or **RRESP**.

2.8.10 XP error handling

Errors are reported at the XP for various reasons.

The following errors are detected in the XP:

- Flit parity error.
- Data check error (DAT channel only).

Flit parity error

Flit parity is generated on a flit upload from a device port to a mesh port, for both internal and external devices. Flit parity check is done on a flit download from a mesh port to a device port.

Flit parity is not generated or checked when a flit is bypassed or looped back across the device ports on the same XP.

Data check error

Data check is enabled in the XP using the DATACHECK_EN parameter.

Data check (Data Byte Parity) bits are generated corresponding to each byte of data on a DAT flit upload from a device port when the corresponding device does not support Datacheck (indicated by DEV_DATACHECK_EN = 0).

Data check is accomplished on a flit download to a device which does not support Data check.

Data check bits are generated and checked when a DAT flit is bypassed or looped back across the device ports on the same XP when the corresponding devices involved do not support Data check.

Error reporting and logging

Flit parity and Data check errors are reported to the RCB. The following table contains flit fields that are logged in the XP configuration register.

Table 2-69 XP configuration register flit fields

Error source	Errstatus					Errmisc			
	DE	CE	MV	UE	V	ERRSRC	SRCID	OPCODE	TGTID
Data Parity P0 REQ channel	1	0	1	0	1	5'b00000	v	v	v
Data Parity P1 REQ channel	1	0	1	0	1	5'b00001	v	v	v
Data Parity P2 REQ channel	1	0	1	0	1	5'b00010	v	v	v
Data Parity P3 REQ channel	1	0	1	0	1	5'b00011	v	v	v
Data Parity P4 REQ channel	1	0	1	0	1	5'b00100	v	v	v
Data Parity P5 REQ channel	1	0	1	0	1	5'b00101	v	v	v
Data Parity P0 RSP channel	1	0	1	0	1	5'b01000	v	v	v
Data Parity P1 RSP channel	1	0	1	0	1	5'b01001	v	v	v
Data Parity P2 RSP channel	1	0	1	0	1	5'b01010	v	v	v
Data Parity P3 RSP channel	1	0	1	0	1	5'b01011	v	v	v
Data Parity P4 RSP channel	1	0	1	0	1	5'b01100	v	v	v
Data Parity P5 RSP channel	1	0	1	0	1	5'b01101	v	v	v
Data Parity P0 SNP channel	1	0	1	0	1	5'b10000	v	v	0
Data Parity P1 SNP channel	1	0	1	0	1	5'b10001	v	v	0
Data Parity P2 SNP channel	1	0	1	0	1	5'b10010	v	v	0
Data Parity P3 SNP channel	1	0	1	0	1	5'b10011	v	v	0
Data Parity P4 SNP channel	1	0	1	0	1	5'b10100	v	v	0
Data Parity P5 SNP channel	1	0	1	0	1	5'b10101	v	v	0
Data Parity P0 DAT channel	1	0	1	0	1	5'b11000	v	v	v
Data Parity P1 DAT channel	1	0	1	0	1	5'b11001	v	v	v
Data Parity P2 DAT channel	1	0	1	0	1	5'b11010	v	v	v
Data Parity P3 DAT channel	1	0	1	0	1	5'b11011	v	v	v
Data Parity P4 DAT channel	1	0	1	0	1	5'b11100	v	v	v
Data Parity P5 DAT channel	1	0	1	0	1	5'b11101	v	v	v
FLIT Parity P0 REQ channel	0	0	1	1	1	5'b00000	v	v	v
FLIT Parity P1 REQ channel	0	0	1	1	1	5'b00001	v	v	v
FLIT Parity P2 REQ channel	0	0	1	1	1	5'b00010	v	v	v
FLIT Parity P3 REQ channel	0	0	1	1	1	5'b00011	v	v	v
FLIT Parity P4 REQ channel	0	0	1	1	1	5'b00100	v	v	v
FLIT Parity P5 REQ channel	0	0	1	1	1	5'b00101	v	v	v
FLIT Parity P0 RSP channel	0	0	1	1	1	5'b01000	v	v	v

Table 2-69 XP configuration register flit fields (continued)

Error source	Errstatus					Errmisc			
FLIT Parity P1 RSP channel	0	0	1	1	1	5'b01001	v	v	v
FLIT Parity P2 RSP channel	0	0	1	1	1	5'b01010	v	v	v
FLIT Parity P3 RSP channel	0	0	1	1	1	5'b01011	v	v	v
FLIT Parity P4 RSP channel	0	0	1	1	1	5'b01100	v	v	v
FLIT Parity P5 RSP channel	0	0	1	1	1	5'b01101	v	v	v
FLIT Parity P0 SNP channel	0	0	1	1	1	5'b10000	v	v	0
FLIT Parity P1 SNP channel	0	0	1	1	1	5'b10001	v	v	0
FLIT Parity P2 SNP channel	0	0	1	1	1	5'b10010	v	v	0
FLIT Parity P3 SNP channel	0	0	1	1	1	5'b10011	v	v	0
FLIT Parity P4 SNP channel	0	0	1	1	1	5'b10100	v	v	0
FLIT Parity P5 SNP channel	0	0	1	1	1	5'b10101	v	v	0
FLIT Parity P0 DAT channel	0	0	1	1	1	5'b11000	v	v	v
FLIT Parity P1 DAT channel	0	0	1	1	1	5'b11001	v	v	v
FLIT Parity P2 DAT channel	0	0	1	1	1	5'b11010	v	v	v
FLIT Parity P3 DAT channel	0	0	1	1	1	5'b11011	v	v	v
FLIT Parity P4 DAT channel	0	0	1	1	1	5'b11100	v	v	v
FLIT Parity P5 DAT channel	0	0	1	1	1	5'b11101	v	v	v

If the device supports Poison (indicated by DEV_POISON_EN = 1), the Datacheck error is factored in the POISON field of the DAT flit. Else, it is factored in as DataError in the RESPERR field.

2.8.11 CXHA error handling

Errors are reported at the CXHA for various reasons.

CXHA uses RAMs as buffers for storing the Read and Write data. The contents of the RAM are protected using byte parity. CXHA reports errors if there is an error that is detected when the contents of the Data RAMs are read. These detected errors are of two types:

- Parity error on *Byte-Enable* (BE) fields of the Write Data RAM
- Parity error on Data and Poison fields of the Read and Write Data RAM.
- Parity error on Data, Poison, and Metadata (if enabled) fields of the Read and Write Data RAM.

Parity error on BE fields of the Write Data RAM

The Write Data buffer RAM stores BE. Parity errors that are detected on BE are treated as UEs. On detecting an error, CXHA does the following:

- Logs the error as UE

Parity error on Data and Poison fields of the Read and Write Data RAM

The Read and Write data buffer RAMs contain the Data and Poison fields. Errors that are detected on these fields are treated as DEs. If an error is detected on Data fields, then the CXHA does the following:

The Read and Write data buffer RAMs contain the Data, Poison, and Metadata (if enabled) fields. Errors that are detected on these fields are treated as DEs. If an error is detected on Data fields, then the CXHA does the following:

- Logs the error as DE
- Poisons the data by setting the corresponding poison bit of the data. For more information about data poisoning, see the *AMBA® 5 CHI Architecture Specification*.

If an error is detected on Poison fields, then the CXHA does the following:

If an error is detected on Poison or Metadata (if enabled) fields, then the CXHA does the following:

- Logs the error as DE
- All Poison bits are set to 1.

2.8.12 CCIX Protocol Error messaging support

CMN-700 CML supports sending of CCIX *Protocol Error* (PER) message to the CCIX Error Agent present on the Host chip.

CMN-700 includes configuration registers, present in CXLA, and a mechanism to trigger a CCIX PER message. It is expected that an external Error Aggregator/Handler present outside CMN-700 collects and consolidates all the errors and uses these registers to trigger a CCIX PER message to the CCIX Error Agent.

CXLA Configuration Registers:

- CCIX PER Message Payload
 - por_cxla_permsg_pyl0_63
 - por_cxla_permsg_pyl0_127
 - por_cxla_permsg_pyl0_128_191
 - por_cxla_permsg_pyl0_192_255
- CCIX PER Message Control
 - por_cxla_permsg_ctl
- CCIX Error Agent ID
 - por_cxla_err_agent_id

Mechanism:

- Error Aggregator external to CMN-700
 - Writes the PER payload in CCIX PER Message Payload registers (por_cxla_permsg_pyl0_*).
 - Sets per_msg_vld_set bit in CCIX PER Message Control register (por_cxla_permsg_ctl). When set, a PER message is sent on the given CCIX link that is determined by the Target ID.

It is the responsibility of CCIX discovery software to program CCIX Error Agent ID in CCIX Error Agent ID por_cxla_err_agent_id register. This programming should happen during initial system bring up and the programmed ID is used as the target ID on CCIX PER message.

PER message is supported only in non-SMP mode and therefore smp_mode_en bit in por_cxla_aux_ctl register must be cleared during initial system startup. For more details on SMP mode, refer to [CML Symmetric Multiprocessor support on page 2-60](#).

PER message is supported only in non-SMP mode. For more details on SMP mode, refer to [CML Symmetric Multiprocessor support on page 2-60](#).

By default, *Error SourceID* (ESID) field from PER message payload (bits[53:48]) is used as source ID on PER message. *per_msg_srcid_ovrd* and *per_msg_srcid* fields in CCIX PER message control register (*por_cxla_permsg_ctl*) can be used to override source ID sent on PER message.

— Note —

CMN-700 CML does not implement a CCIX Error Agent. It can accept the incoming PER messages, but these messages are dropped at CXLA.

2.9 Transaction handling

The handling of certain CHI transaction types and fields differs according to the CMN-700 device type.

Some devices fully support certain transaction types or fields, whereas others do not do any processing of those transactions. Furthermore, some transaction types are unsupported, such as barriers.

This section contains the following subsections:

- [2.9.1 Atomics on page 2-195](#).
- [2.9.2 Exclusive accesses on page 2-196](#).
- [2.9.3 Barriers on page 2-197](#).
- [2.9.4 DVM messages on page 2-197](#).
- [2.9.5 Completer Busy indication on page 2-199](#).
- [2.9.6 MTSX functionality on page 2-204](#).
- [2.9.7 REQ RSVDC propagation on page 2-205](#).
- [2.9.8 DAT RSVDC propagation on page 2-205](#).

2.9.1 Atomics

CMN-700 supports atomic accesses to both cacheable and non-cacheable memory locations.

————— Note ————

Atomics are not supported at the RN-I interfaces.

Atomic requests in HN-F

The HN-F completes all CHI atomic requests that it receives, both for Cacheable and Non-cacheable transactions.

For Cacheable transactions, the HN-F completes any appropriate coherent actions and, if necessary, obtains the targeted cache line from memory. The HN-F then completes the required atomic operation and issues the appropriate response, with or without data.

For Non-cacheable transactions, the HN-F does not send an atomic request to the SN. As the final PoS/PoC for all memory traffic, the HN-F is able to issue a read to the SN, atomically update the copy of the data in the HN-F, and then write back the result to the SN. This approach means that the SN never receives CHI atomic requests, as the HN-F completely handles the requests.

Atomic requests in SN

The SN node (CHI memory controller or SBSX bridge) does not process atomic requests.

Atomic requests in HN-I

The HN-I does not complete atomic transactions.

On receiving an atomic request, the HN-I generates an appropriate error response to the originating master.

Atomic requests in RN-I and RN-D

RN-I and RN-D support atomic transactions in CMN-700. These nodes can receive atomics from ACE5-Lite and AXIS5 masters, and translate them on CHI before sending them to HN-F, HN-I, or CXRH nodes.

Atomics and write transactions share a write tracker for processing in RN-I and RN-D. There is a separate *Read Data Buffer* (RDB) for atomic responses. The **NUM_ATOMIC_BUF** parameter determines the depth of this buffer.

————— **Note** —————

For atomic transactions arriving at RN-I and RN-D from ACE5-Lite and AXI5 masters, all write strobes within **AWSIZE** must be set. RN-I and RN-D do not allow sparse write strobes for atomic transactions.

2.9.2 Exclusive accesses

CMN-700 supports exclusive accesses to both Shareable and Non-shareable locations.

For more information, see the *AMBA® 5 CHI Architecture Specification*.

Exclusive accesses in HN-F

The HN-F supports exclusive access on ReadNoSnp, WriteNoSnp, ReadShared, ReadClean, ReadNotSharedDirty, and CleanUnique transactions to any address that maps to the HN-F.

RNs generate ReadNoSnp and WriteNoSnp Exclusives for memory locations that are marked Non-cacheable or Device. ReadShared, ReadClean, ReadNotSharedDirty, and CleanUnique exclusives are used for shareable and coherent memory locations. Each HN-F partition includes 64 exclusive monitors for tracking of these transaction types. Each monitor can act as both a PoC monitor and System monitor, as defined by the *AMBA® 5 CHI Architecture Specification*.

Only 64 unique logical threads, which are designated by a unique combination of SrcID and LPID, can concurrently access the HN-F exclusive monitors.

Exclusive accesses in HN-I

HN-Is support exclusive access on ReadNoSnp and WriteNoSnp transactions to any address that maps to an HN-I.

Each HN-I partition includes 32 exclusive monitors as defined in the *AMBA® 5 CHI Architecture Specification* for tracking of these transaction types. Only 32 unique logical threads can concurrently access the HN-I system exclusive monitors. These threads can be either processor or device threads, and are designated by a unique combination of SrcID and LPID.

Each HN-I partition includes exclusive monitors, as defined in the *AMBA® 5 CHI Architecture Specification*, for tracking of these transaction types. The number of monitors is specific to the configuration and is determined by the number of RN-Fs, RN-Is, and RN-Ds in the configuration. The number of monitors determines the number of unique logical threads that can concurrently access the HN-I system exclusive monitors. These threads can be either processor or device threads, and are designated by a unique combination of SrcID and LPID.

————— **Note** —————

HN-P does not support exclusive accesses.

All exclusives targeting the HN-I are terminated at the HN-I and are not propagated downstream. Exclusives are terminated regardless of the value of the HN-I PoS control register and auxiliary control register.

CML support for exclusive accesses

CMN-700 CML supports exclusive accesses in some circumstances. Support for these transactions and the guidance for configuration depends on whether SMP mode is enabled.

CML_SMP mode

In SMP mode, CMN-700 CML supports remote exclusive accesses from RN-Fs.

Remote exclusive accesses from an RN-I or RN-D are not supported.

Support for remote exclusive accesses include these constraints:

- CXRA in local CCG block passes Excl and LPID fields of incoming CHI request on request message USER (Ext) field.
- CXHA in the remote CCIX gateway (CXG) block extracts these bits from request message USER (Ext) field. CXHA then sends these bits on respective CHI Excl and LPID fields. CXHA sets the source type as RN-F based on its RAID to LDID register.
- Exclusive OK (EXOK) response is sent as `0b01` on RespErr field.

HN-Fs and HN-Is monitor exclusives from remote RN-Fs using existing exclusive monitors. To track remote exclusives, the monitors track the source type, HA_LOGICAL_ID, LDID, and LPID fields of the incoming request.

CXG/CCIX1.1 exclusives

We recommend setting the `lnk<X>_excl_load_dwngrd` and `lnk<X>_excl_store_dwngrd` bits in the respective `por_cxg_ra_cxprtcl_link<X>_ctl` register. If these bits are set, a shareable exclusive access, for example a load or store access, is downgraded to a shareable non-exclusive access. In other words, the Excl attribute is dropped from the access. The access is then sent on the corresponding CCIX link. Any incoming OK response is converted to EXOK when sent to the requesting CHI RN.

Exclusive accesses are not supported to Normal non-cacheable or Device memory. Load exclusives are sent as Normal loads and the incoming OK response is passed to the requesting RN. Store exclusives are terminated at CXRA with an NDE response.

CXRA also has more bits to override the RespErr field on response for an exclusive access in non-SMP mode. The `lnk<X>_excl_resperr_value` value in the `por_cxg_ra_cxprtcl_link<X>_ctl` register can override incoming OK responses. You must take care when using these bits, because response overrides are not expected for normal use.

Exclusive accesses in RN-I and RN-D

RN-I supports up to two active exclusive threads at any given AXI port. To differentiate the exclusive threads, RN-Is provides a per port 11-bit mask to extract the bit from **AxID**.

The 11-bit mask `por_{rni,rnd}_s<X>_port_control` can be found in the respective RN-I and RN-D AXI port control registers.

2.9.3 Barriers

Barriers were deprecated from CHI-B onwards. All masters (fully coherent and I/O coherent) must handle barriers at the source.

When memory barrier ordering or completion guarantees are required, masters must wait for the responses from all required previous transactions that are issued into the interconnect. No barrier requests can be issued into the interconnect.

All requesting devices that are attached to an interconnect must have a configuration option or strap that prevents issuing of any barriers. If a barrier is issued into the interconnect, the results are unpredictable.

Note

The DVM_SYNC command, the DVM synchronization that might be initiated by an Arm DSB instruction, is sent to the DVM block, and executes appropriately.

For more information about barriers, see the *AMBA® 5 CHI Architecture Specification* and the *AMBA® AXI and ACE Protocol Specification*.

2.9.4 DVM messages

If an RN-F supports *Distributed Virtual Memory* (DVM) messages, it can send DVM requests and receive DVM snoops.

The DVM Node (DN) in CMN is responsible for handling DVM messages. CMN supports up to 4 DNs per chip. DNs are present in HN-D, HN-T and HN-V devices.

The minimum number of DNs is 1 using HN-D, and the maximum number of HND + HNT + HNV is 4.

Each DN is part of a DN domain, a DN domain is defined as follows:

- A group of RNs and CXRAs and an associated DN
- The RNs and CXRAs assignment is achieved by configuring XPs to a DN domain, when configuring the mesh.
- A DN domain must be built using contagious XPs
- A DN domain must contain only one DN: HN-D, HN-T, or HN-V.
- All the RNs and CXRAs in a DN domain send DVM messages (OPs and Syncs) to the associated DN.

A DVM message from an Rn-F is sent to DN in the RNF's DN domain. On receiving the DVM message, the DN:

- Forwards the DVM message as a snoop to the participating RNs and CXRAs in its DN domain
- Forwards the DVM message to peer DNs in the other DN domains
 - The Peer-DNs snoop the participating RNs and CXRAs in their respective DN domains
 - Collect the Snoop responses
 - Send a single Snoop response to the initiating DN
- Collects the individual snoop responses.
- Sends a single response back to the RNF that originated the DVM message transaction

The default RNs and CXRAs association done while configuring the XP can be reconfigured at boot time using registers:

Each DN has registers to associate the RNs and CXRA within the DN domain (por_dn_domain_rnf0, por_dn_domain_rnf1, por_dn_domain_rnf2, por_dn_domain_rnf3, por_dn_domain_rnd, por_dn_domain_cxra), and each RN and CXRA has a configurable register to associate the DVM Node in its DN domain (dn_nodeid in rnsam_status).

The **SYSCOREQ/SYSCOACK** mechanism provides proxy snoop responses in scenarios when the RN is powered down. For more **SYSCOREQ/SYSCOACK** information, refer to [2.3.9 RN entry to and exit from Snoop and DVM domains on page 2-92](#).

An RN that issues DVM messages must also be able to receive DVM messages. If this requirement is violated, the system must not rely on the DVM message causing any DVM snoops.

- An RNF can issue only one outstanding DVMOp (Sync).

For more information about DVM messages, see the *AMBA® 5 CHI Architecture Specification*.

Support for early completion of DVMOp requests

CMN-700 HN-D and CXRA nodes can give early completions for DVMOp requests. You enable or disable this mode with programmable register bits.

The following programmable bits enable or disable this mode in these nodes:

HN-D	disable_dvmop_early_comp bit in por_dn_aux_ctl register
CXRA/CCRA	dvm_earlycomp_en bit in por_<cxg/ccg>_ra_aux_ctl register

By default, the early DVMOp completion mode is disabled at HN-D. When you enable early DVMOp completion, the following errors are not reported as NDE on Comp:

- Poison, DataCheck, and Data Error on RXDATFLIT
- NDE on Snoop responses

CXRA can give early completions for DVMOp requests that are sent over a CCIX SMP link. By default, this mode is enabled at CXRA. When this mode is enabled, any NDE on DVMOp completion is dropped.

2.9.5 Completer Busy indication

Transaction completers can use the Completer Busy (CBusy) field to indicate their current level of activity. RNs use this indication to determine whether to throttle outgoing traffic.

CMN-700 implements the CBusy indication function in the following node types:

- HN-F
- SBSX
- MTSX
- CXRA

HN-I, HN-T, HN-D, and HN-P always drive the CBusy values as `0b000`.

HN-F CBusy

HN-F uses the *Point-of-Coherency Queue* (POCQ) occupancy level to indicate its current activity level. The following table shows the default CBusy values for a 32 entry POCQ. These values represent the default HN-F CBusy response behavior to RNs.

Table 2-70 HN-F POCQ CBusy thresholds for 32e POCQ

CBusy[2:0]	Tracker occupancy level
<code>0b011</code>	≥ 24
<code>0b010</code>	≥ 16
<code>0b001</code>	≥ 8
<code>0b000</code>	< 8

HN-F also supports a CBusy multi-source mode. The CBusy[2] bit indicates that multiple sources, or RNs, have outstanding requests pending in the HN-F POCQ. HN-F also supports a mode where, when calculating CBusy[2], it excludes outstanding RN-I requests in the POCQ. This mode is enabled if the `hnf_cbusy_mtbit_exclude_rni` field of the `por_hnf_cbusy_limit_ctl` register is set to `0b1`. If this field is set, then HN-F ignores outstanding requests from the RN-I while calculating the CBusy[2] value.

SBSX and MTSX CBusy

SBSXs and MTSXs only drive CBusy on returning TXDAT flits targeting RNs. These nodes use two hierarchical trackers to drive the CBusy field: ReqTracker and DART. The CBusy field reflects the occupancy levels of both trackers combined. Similar to HN-F, the activity thresholds are programmable. The following table shows the default occupancy threshold for 96 entry trackers.

Table 2-71 SBSX and MTSX tracker CBusy thresholds

CBusy[2:0]	Tracker occupancy level
<code>0b011</code>	≥ 72
<code>0b010</code>	≥ 48
<code>0b001</code>	> 24
<code>0b000</code>	< 24

SBSXs and MTSXs do not use the multi-source mode bit, so CBusy[2] is always set to `0b0`.

CXRA CBusy

CXRA uses the request tracker (RHT) occupancy level to indicate the current activity level. The following table shows the default values for a 256 entry RHT. This behavior is the default mode of the CXRA outgoing CBusy in all responses to RNs.

Table 2-72 CXRA RHT CBusy thresholds

CBusy[2:0]	Tracker occupancy level
0b011	≥ 192
0b010	≥ 128
0b001	≥ 64
0b000	<64

CXRAs do not use the multi-source mode bit, so CBusy[2] is always set to 0b0.

Advanced CBusy handling in HN-F

CMN-700 HN-F supports advanced CBusy handling and request throttling to SN-F.

HN-F to RN CBusy handling

The responses that are sent from HN-F to RN through RSP and DAT channels carry CBusy values. HN-F has multiple different modes to determine how the CBusy values are specified in the response messages.

HN-F can be configured to respond to RNs with a CBusy value that reflects one of the following options:

- Total number of outstanding requests in the HN-F POCQ (default mode).
- Independent CBusy values for reads and writes:

CompData type requests (All Read* requests)

CBusy value is based on number of outstanding reads in the POCQ.

Comp type requests (Writes, Evict, atomics, CMOs)

CBusy value is based on number outstanding writes in the POCQ.

- Return SN-F CBusy value instead of returning value that is based on HN-F POCQ:
 - Read requests receive the Read CBusy of the SN-F.
 - Write requests receive the Write CBusy of the SN-F.
- Return whichever CBusy value is the highest between HN-F POCQ and SN-F.

Comp type requests can be further filtered into the following categories:

- CopyBack type requests (Evict, WriteClean, WriteEvictFull, or WriteBack*).
- NonCopyBack type requests (including WriteNoSnp*, WriteUnique*, Combined Write, (P) CMO's, and atomics).

Write filtering of CopyBack versus NonCopyBack types is only supported when you configure HN-F to respond with the CBusy of the POCQ. Write filtering is not supported if the HN-F returns the CBusy value of the SN-F.

The following table shows the format of the por_hnf_cbusy_limit_ctl_register. This register controls the HN-F CBusy threshold for Read requests.

Table 2-73 por_hnf_cbusy_limit_ctl register for CBusy thresholds (all requests or read types)

Bits	Name	Description
[7:0]	hnf_cbusy_low_limit	Specifies the POC valid threshold at which HN-F is considered least busy.
[15:8]	hnf_cbusy_med_limit	Specifies the POC valid threshold at which HN-F is considered medium busy.
[23:16]	hnf_cbusy_high_limit	Specifies the POC valid threshold at which HN-F is considered very busy.

Table 2-73 por_hnf_cbusy_limit_ctl register for CBusy thresholds (all requests or read types) (continued)

Bits	Name	Description
[48]	hnf_cbusy_rd_wr_types_en	Allows separate CBusy values for reads versus writes. When enabled, the thresholds in this register are only applicable to read type requests. Otherwise these values are the default thresholds for calculating CBusy for all request types in POCQ of the HN-F. This bit must be set when sn_cbusy_prop_en = 0b1 to propagate the SN CBusy.
[63]	hnf_cbusy_mtbit_exclude_rni	Allows HN-F to ignore outstanding read requests from RN-I when calculating busyness.

The following table shows the format of the por_hnf_cbusy_write_limit_ctl register. This register controls the HN-F CBusy threshold for Write requests.

Table 2-74 Register for CBusy thresholds (write requests)

Bitfield	Field	Description
[7:0]	hnf_cbusy_low_limit	Specifies the POC valid threshold at which HN-F is considered least busy.
[15:8]	hnf_cbusy_med_limit	Specifies the POC valid threshold at HN-F is considered medium busy.
[23:16]	hnf_cbusy_high_limit	Specifies the POC valid threshold at which HN-F is considered very busy.
[48]	hnf_cbusy_sep_copyback_types	When set, HN-F calculates CBusy based on outstanding CopyBack and NonCopyBack type requests independently in the HN-F POCQ.

The following table shows the CBusy values that are returned to RNs according to programming.

Table 2-75 HN-F CBusy value propagation according to programming

hnf_adv_cbusy_mode_en	hnf_cbusy_rd_wr_types_en	sn_cbusy_prop_en	cbusy_highest_of_all_en	CBusy value passed to RN
0b0	x	x	x	POCQ CBusy value is returned.
0b1	0b0	x	x	POCQ CBusy value is returned.
0b1	0b1	x	x	POCQ CBusy value for read or write is returned, according to the request type.

Table 2-75 HN-F CBusy value propagation according to programming (continued)

hnf_adv_cbusy_mode_en	hnf_cbusy_rd_wr_types_en	sn_cbusy_prop_en	cbusy_highest_of_all_en	CBusy value passed to RN
0b1	0b1	0b1	0b0	SN CBusy value for read or write is returned for the corresponding SN group, according to the request type.
0b1	0b1	x	0b1	Highest of either the SN or POCQ CBusy value for read or write is returned, according to the request type.

Where applicable, HN-F returns the read or write CBusy value according to opcode type.

Write CBusy values can be further separated into CopyBack and NonCopyBack values using the hnf_cbusy_sep_copyback_types field. This separation only applies when HN-F is programmed to propagate the POCQ CBusy values. In this mode, CopyBack write type values account for pending WriteClean*, WriteBack*, WriteEvictFull*, and Evict type operations. NonCopyBack write type values account for all other pending write operations (WriteUnique*, WriteNoSnp*). Combined Write* and (P)CMO operations are counted towards NonCopyBack types. Standalone CMOs are not counted towards either of the CopyBack or NonCopyBack type requests.

HN-F to SN-F CBusy based throttling

HN-F can identify two groups of memory controllers using a configuration bit for each SN. These groups are known as Group A or Group B. You can use the two groups to identify fast and slow memory types. Therefore, the HN-F can handle traffic to and from the two types independently of each other. HN-F can track the read and write busyness to each SN-F group over a configurable transaction window. It can be programmed to track the last 128 or 256 transactions. When HN-F has received as many responses from SN-F, it measures the current busyness for each group of SN and request types (read and write). The measured busyness is then used to throttle the traffic to SN-F appropriately. The threshold for measuring the CBusy for the last 128 or 256 transactions is also configurable. For example, if HN-F receives ≥ 16 transactions and CBusy = 0b11 from Group 0 SN-Fs, then HN-F treats the SN-F CBusy value as 0b11. This value corresponds to very busy.

The threshold for measuring the CBusy for the last 128 or 256 transactions is also configurable. For example, consider a scenario where HN-F is programmed to calculate the last 128 CBusy responses. HN-F tracks the number of times it receives CBusy values of 0b00, 0b01, 0b10, and 0b11 for each SN group. In this example, the HN-F receives more than 16 CBusy = 0b11 responses from Group 0 SN-Fs out of the last 128 responses. In this case, HN-F treats the final SN-F CBusy value as 0b11 for the subsequent 128 transactions while continuing to accumulate new CBusy response values.

HN-F can be configured to throttle outgoing requests in either a static mode or a default dynamic mode:

- Static throttling mode: HN-F controls the fixed number of transactions outstanding at any point for a given SN group and request type:
 - CBusy = **0b11** (Very busy): HN-F restricts transactions to a maximum of a quarter of the number of POCQ entries.
 - CBusy = **0b10** (Medium busy): HN-F restricts transactions to a maximum of half of the number of POCQ entries.
 - CBusy = **0b01** (Low busy): HN-F restricts transactions to three quarters of the number of POCQ entries.
 - CBusy = **0b00** (Not busy): HN-F can issue as many requests as the number of POCQ entries.
- Dynamic throttling mode: The number of *Outstanding Transactions* (OTs) can be dynamically throttled according to programmed values. It can be configured to increment or decrement the transaction count by two, four, or eight transactions after every 128 or 256 transaction window (as programmed).
 - CBusy = **0b11** (Very busy): Decrement the OT count.
 - CBusy = **0b10** (Medium busy): No change to the current OT count.
 - CBusy = **0b01** (Low busy): Increment the OT count.
 - CBusy = **0b00** (Not busy): Increment the OT count.

When you configure an HN-F to respond to RNs with the CBusy value of an SN-F, HN-F can propagate the CBusy value according to the SN-F group that the request targets. For example, consider an RN-F sending a read request that is targeting SN group A. The RN can receive the CBusy value for a group A SN, even if the request hits in SLC and therefore the HN-F completes the request.

The following table shows the format of the por_hnf_cbusy_resp_ctl register. This register controls the CBusy responses.

Table 2-76 por_hnf_cbusy_resp_ctl register for configuring CBusy value on responses

Bits	Name	Description: Controls the CBusy responses
[0]	sn_cbusy_prop_en	When set to 0b1 , HN-F responds with the CBusy values from SN-F instead of using its own POCQ occupancy-based thresholding. Read and write modes are still controlled using por_hnf_cbusy_limit_ctl and por_hnf_cbusy_write_limit_ctl.
[4]	Cbusy_highest_of_all_en	When set to 0b1 , HN-F responds with the CBusy values from the highest of group A and group B.
[7]	cbusy_sn_static_ot_mode_en	Enables the static OT throttling to SN.
[21:16]	cbusy_sn_dynamic_ot_count	Count by which the OT count is incremented or decremented for dynamic OT throttling.

The following table shows the format of the por_hnf_sam_sn_properties1 register. This register controls the group to which each SN belongs.

Table 2-77 Per SN group identifier in por_hnf_sam_sn_properties registers

Field	Description
sn0_group	0b0 Group A
sn1_group	
sn2_group	
sn3_group	
sn4_group	
sn5_group	
sn6_group	
sn7_group	
Region0_sn_group	0b1 Group B
Region1_sn_group	

The following table shows the format of the por_hnf_cbusy_sn_ctl register. This register controls the CBusy sampling.

Table 2-78 por_hnf_cbusy_sn_ctl register for CBusy sampling control

Bitfield	Field	Description
[9:0]	hnf_cbusy_threshold_cntr01	CBusy threshold at which SN-F is considered busy for Counter_01.
[25:16]	hnf_cbusy_threshold_cntr10	CBusy threshold at which SN-F is considered busy for Counter_10.
[41:32]	hnf_cbusy_threshold_cntr11	CBusy threshold at which SN-F is considered busy for Counter_11.
[56:48]	hnf_cbusy_txn_cnt	Number of SN responses over which the CBusy counters are tracked.

HN-F continues to propagate the multi-source bit (CBusy[2]) in the advanced modes.

2.9.6 MTSX functionality

The MTSX device is used when you require support for *Memory Tagging Extensions* (MTE), but an AXI slave device in your system does not support MTE. Internally, MTSX includes all SBSX functionality and support for MTE using a *Memory Tag Unit* (MTU).

MTSX can generate separate data and tag requests on the AXI interface according to the CHI request. For the MTSX to generate tag requests, you must program the MTSX address generation registers. For more information about programming these registers, see [3.4.7 MTSX programming on page 3-1245](#).

Optionally, the MTSX can include a *Tag Cache* (TC) to store tags locally. This configuration allows for faster tag access and reduced AXI traffic. If the TC is not included, each CHI request requiring tag access causes the MTSX to generate two AXI requests. One request is for data and one request is for tag access.

Software-configurable error injection in MTSX

The MTSX supports software-configurable error injection and reporting. This feature enables testing of the software error handler routine for TC double-bit ECC data errors.

The por_mtu_err_inj configuration register enables configurable error injection and reporting for a given PA value.

Any read for which the TC provides the data is defined as a TC hit. If error injection and reporting are enabled, any TC hit drives the data double-bit error and a fault interrupt through the RAS control block

for the TC hit. This functionality emulates a double-bit ECC error in the TC data RAM but does not pollute the TC data RAM through the fill path.

— Note —

This mechanism is designed to mimic TC data ECC errors for TC hits. If enabled, the mechanism only causes an error to be logged and, optionally, an interrupt to be generated. TC misses do not drive any errors or error interrupts.

For more information about configuring error injection, see [por_mtu_err_inj](#) on page 3-432.

Hardware-based TC flush engine in MTSX

The MTSX supports a flush engine mechanism to flush the TC. The flush engine ensures that all cache lines in the TC are flushed from CMN-700.

Two configuration registers per MTSX instance support the TC flush engine:

por_mtu_tc_flush_pr The TC flush policy register. This register triggers a flush to start and indicates the flush operation type.

por_mtu_tc_flush_sr The TC flush status register. This register indicates flush completion.

A flush request is initiated by writing to the por_mtu_tc_flush_pr. After a flush request has been initiated, the por_mtu_tc_flush_pr register should not be written again until the flush completes.

When all cache lines are flushed, the mtu_tc_flush_complete bit in the por_mtu_tc_flush_sr register is set. Setting this bit indicates that the flush engine has completed its operation.

The flush engine has two modes of operation:

CleanInvalid

The default mode. The flush engine writes back any modified data to memory before invalidating a cache line.

CleanShared

The flush engine writes any modified data back to memory but keeps a clean copy of the cache line in the TC.

The mtu_tc_flush_mode bit of the por_mtu_tc_flush_pr register sets the mode of operation for a flush.

For more information about configuring TC flush, see the following register descriptions:

- [por_mtu_tc_flush_pr](#) on page 3-406
- [por_mtu_tc_flush_sr](#) on page 3-407

2.9.7 REQ RSVDC propagation

CMN-700 supports propagation of the RSVDC field of the CHI REQ flit through the interconnect.

For a multi-chip system, the REQ RSVDC field is preserved and only passed over a CCIX SMP link. For AXI slave interfaces, the incoming **AxUSER** field is mapped to CHI REQ RSVDC field and is propagated through the interconnect. For AXI master interfaces, CHI REQ RSVDC field is mapped to **AxUSER**. This field is not stored in SLC and so is not preserved for requests that are allocated in SLC.

2.9.8 DAT RSVDC propagation

CMN-700 supports propagation of the RSVDC field of the CHI DAT flit through the datapath and SLC for full cache line read and write operations from RN-Fs to SN-Fs. This support includes CCIX traffic when in SMP mode.

This feature is enabled by setting the **META_DATA_EN** parameter to 1. For requests that are allocated in SLC, DAT RSVDC is also stored in the SLC.

The RSVDC is not preserved for AXI traffic that RN-I or RN-D initiate, or targeting SBSX, HN-I, HN-T, HN-P, or HN-D. It is also not preserved for non-SMP CXRA, CXHA, or CXSA traffic.

The **RUSER** and **WUSER** signal widths increase on ACE-Lite master and slave interfaces when **META_DATA_EN** is set to 1. However they are not used to propagate DAT RSVDC values.

———— **Note** ————

Partial cache states are not supported when the **META_DATA_EN** parameter is set.

———— **Note** ————

WriteNoSnpZero and WriteUniqueZero opcodes do not update DAT RSVDC value.

———— **Note** ————

DAT RSVDC propagation and storage in SLC is only available if MTE_EN is 0.

2.10 Processor events

CMN-700 supports communicating processor events to all processors in the system.

Refer to the processor event interface signals described in [B.13 Processor event interface signals on page Appx-B-1398](#).

When a processor generates an output event that is triggered by an SEV instruction, it is broadcast to all processors in the system. Similarly, anytime an exclusive monitor within HN-F or HN-I is cleared, an output event is broadcast to all processors in the system. The event interface signals are also present at RN-I and RN-D nodes, for use by components such as a *System Memory Management Unit* (SMMU).

The logical operator OR is used to combine the EVENT signals, then the result is broadcast to the processors.

2.11 Quality of Service

CMN-700 includes end-to-end QoS capabilities which support latency and bandwidth requirements for different types of devices.

The QoS device classes are:

Devices with bounded latency requirements

These devices are primarily real-time or isochronous that require some or all of their transactions complete within a specific time period to meet overall system requirements. These devices are typically highly latency-tolerant within the bounds of their maximum latency requirement. Examples of this class of device include networking I/O devices and display devices.

Latency-sensitive devices

The performance of these devices is highly impacted by the response latency that is incurred by their transactions. Processors are traditionally highly latency-sensitive devices, although a processor can also be a bandwidth-sensitive device depending on its workload.

Bandwidth-sensitive devices

These devices have a minimum bandwidth requirement to meet system requirements. An example of this class of device is a video codec engine, which requires a minimum bandwidth to sustain real-time video encode and decode throughput.

Bandwidth-hungry devices

These devices have significant bandwidth requirements and can use as much system bandwidth as is made available, to the limits of the system. These devices determine the overall scalability limits of a system, with the devices and system scaling until all available bandwidth is consumed.

Note

A device can be classified into one or more of these classes, depending on its workload requirements.

Support for these different types of devices and their resulting traffic is included in the AMBA 5 CHI protocol and in the entirety of CMN-700 microarchitecture. Each component in CMN-700 contributes to the overall QoS microarchitecture.

This section contains the following subsections:

- [2.11.1 Architectural QoS support on page 2-208](#).
- [2.11.2 Microarchitectural QoS support on page 2-208](#).
- [2.11.3 QoS configuration example on page 2-212](#).

2.11.1 Architectural QoS support

The AMBA 5 CHI protocol includes a 4-bit *QoS Priority Value* (QPV) with all message flits.

The QPV of the originating message must propagate for all messages in a transaction. The QPV is defined as higher values being higher priority and lower values being lower priority. All CMN-700 components use the QPV to provide prioritized arbitration and to prevent Head-of-line blocking based on the QPV.

2.11.2 Microarchitectural QoS support

The QPV of RN requests must be modulated depending on how well or poorly their respective QoS requirements are met.

QoS regulators

Although the QoS-modulation capability can be integrated into the RN, CMN-700 enables system designers to include non-QoS-aware devices in the CMN-700 system, but still have these devices meet the QoS-modulation requirements of the CMN-700 QoS microarchitecture.

CMN-700 includes inline QoS regulators that perform QoS modulation without requiring any QoS-awareness by the requesting device. A QoS regulator introduces an interstitial layer between an RN and the interconnect that monitors whether the bandwidth and latency requirements of the RN are being met. It also performs in-line replacement of the RN-provided QPV field as required, adjusting upwards to increase priority or downwards to reduce priority in the system.

The QoS regulators are present at all entry points into CMN-700:

- For CHI ports, the regulator is present in the XP.
- For ACE-Lite/AXI4 slave interfaces, the regulator is present at the ACE-Lite/AXI4 side of the protocol bridge.

CMN-700 QoS regulators have three operating modes:

- Pass-through.
- Programmed QoS value.
- Regulation.

These operating modes are controlled through memory-mapped configuration registers.

QoS regulator operation

The values of the base QPV, **AxQOS** for AXI and ACE-Lite interfaces or **RXREQFLIT.QOS** for CHI ports, are inputs to the QoS subblock.

When latency regulation or period regulation is enabled, the values generated by the regulators replace the base QPV values. For an RN-F, a single QoS regulator monitors CHI transactions that return data to the RN-F such as reads, atomics, and snoop stash responses. The regulated QPV is applied to all CHI requests from that RN-F. For an RN-I or RN-D, separate QoS regulators exist for AR and AW channels.

The QoS regulators can operate in either latency regulation mode or period regulation mode. The registers to configure the QoS regulators exist in each RN-I, RN-D, and XP.

Latency regulation mode

When configured for latency regulation, the QoS regulator increases the QPV whenever actual transaction latency is higher than the target, and decreases the QPV when it is lower:

- For every cycle that the latency of a transaction is more than the target latency, the QPV increases by a fractional amount. The scale factor K_i determines this amount.
- For every cycle that the latency of a transaction is less than the target latency, the QPV decreases by the same fractional amount. The scale factor K_i determines this amount.

The QoS Latency Target register specifies the target transaction latency in cycles.

The QoS Latency Scale register specifies the scale factor K_i . It is coded in powers of two, so that a programmed value of $0x0 = 2^{-3}$ and a programmed value of $0x7 = 2^{-10}$.

The QoS regulator can be programmed to operate in latency regulation mode by programming the following bits in the QoS Control register:

- Set the qos_override_en bit to 0b1.
- Set the lat_en bit to 0b1.
- Set the reg_mode bit to 0b0.
- Set the pqv_mode bit to 0b0.

Period regulation mode for bandwidth regulation

When configured for period regulation, the QoS regulator increases the QPV whenever the period between transactions is larger than the target, and decreases the QPV when it is lower:

- For every cycle that the period between transactions (as measured at dispatch time) is more than the target period, the QPV increases the scale factor K_i by a fractional amount.
- For every cycle that the period between transactions is less than the target period, the QPV decreases the scale factor K_i by the same fractional amount.

The QoS Latency Target register specifies the target period in cycles.

The QoS Latency Scale register specifies the scale factor K_i . It is coded in powers of two, so that a programmed value of $0x0 = 2^{-3}$ and a programmed value of $0x7 = 2^{-10}$.

The QoS regulator can be programmed to operate in period regulation mode by programming the following bits in the QoS Control register:

- Set the qos_override_en bit to 0b1.
- Set the lat_en bit to 0b1.
- Set the reg_mode bit to 0b1.

There are two modes of period regulation:

Normal mode

The QPV does not increase or decrease when there are zero outstanding transactions.

Quiesce high mode

The QPV increases by a fractional amount, which the scale factor K_i determines, in every cycle where there are zero outstanding transactions.

The mode of period regulation can be selected by programming the pqv_mode bit in the QoS Control register.

RN-I and RN-D bridge QoS support

In addition to the QoS regulators, the RN-I and RN-D bridge provides QoS-aware arbitration mechanisms.

To simplify arbitration logic, all transactions are split into two QoS Priority Classes (QPCs), high and low. QoS-15 transactions make up the high class. All other transactions are considered to be in the low class.

Port multiplexer arbitration

An RN-I and RN-D bridge includes three ACE-Lite and ACE-Lite-with-DVM ports. The RN-I and RN-D bridge selects between these ports for allocation into its transaction tracker. This selection process makes the allocated transaction a candidate for issuing to a Home Node. The port multiplexer is arbitrated using the following strategy:

- High QPC first, then the low QPC.
- Round-robin arbitration among the AMBA ports within a QPC.

Tracker allocation

When transactions are allocated into the tracker, they are scheduled for issuance to a Home Node based on QPC. This strategy is the same strategy as port mux arbitration.

- High QPC first, then the low QPC.
- Round-robin arbitration in a QPC among the transactions for issue.

HN-F QoS support

The HN-F interprets the 4-bit QPV at a coarser granularity. Following table shows default configuration. For more information, please refer to section xx.yy (HN-F Class based Resource allocation and arbitration).

The HN-F includes the following QoS support mechanisms, if configured to use QoS based classes:

QoS decoding in HN-F

The HN-F interprets the 4-bit QPV at a coarser granularity, as the following table shows. For more information, refer to section [4.5 HN-F class-based resource allocation and arbitration on page 4-1295](#)

————— Note ————

Table below has default configuration and is software-programmable.

Table 2-79 QoS classes in HN-F

QoS value	Class	Dedicated	Contended Min	Max Allowed
15	Class 0	0	POCQ_ENT / 4	POCQ_ENT - 1
14-12	Class 1	0	POCQ_ENT / 4	POCQ_ENT - 2
11-8	Class 2	0	POCQ_ENT / 4	POCQ_ENT / 2
7-0	Class 3	0	POCQ_ENT / 4	POCQ_ENT / 8

QoS class and POCQ resource availability

The HN-F includes 16, 32, 64, or 128 entry structure, the Point-of-Coherency Queue (POCQ), from which all transaction ordering and scheduling is performed. The POCQ buffers are shared resources for all QoS classes, with one entry being reserved for internal use. POCQ is partitioned so that different classes can use configurable number of entries as dedicated, max_allowed or contended_min as shown in figure earlier, ensuring bandwidth and latency requirements of higher priority transactions are met. For more information, please refer to section [4.5 HN-F class-based resource allocation and arbitration on page 4-1295](#)

HN-I, SBSX, and MTSX QoS support

The HN-I bridge provides QoS-aware arbitration mechanisms for static grants and AMBA requests.

To simplify arbitration logic, all transactions are split into two QPCs: high and low. QoS-15 transactions make up the high class. All other transactions are considered to be in the low class.

————— Note ————

SBSX and MTSX QoS support is identical to that of the HN-I.

Dynamic credit tracker allocation

Requests allocate into the tracker until it is full, after which requests are then retried and the HN-I increments a credit counter for the affected RNs in an internal retry bank.

When a tracker entry is cleared, the HN-I checks the retry bank for any retried transactions. If any are present, the newly available tracker entry is reserved and a static credit grant is sent to an RN chosen using the following algorithm:

- Choose an RN marked as high QPC first, then the low QPC.
- Use round-robin arbitration within a QPC.

Scheduling to AMBA interface

When transactions are allocated into the tracker, they are scheduled for issue on the AMBA interface based on QPC following a similar strategy to static credit allocation:

- Choose high QPC first, then the low QPC.
- Use round-robin arbitration within a QPC.

Write data buffers are also allocated based on QPC class. For write requests that are ready to issue:

- Choose high QPC first, then the low QPC.
- Use round-robin arbitration within a QPC.

CML QoS overrides

You can program CMN-700 to override the QPV on incoming transactions through the HA.

You can program the HA config control register, [*por_cxg_ha_cfg_ctl* on page 3-586](#), to hold a QoS override value. The HA overrides the QPV value on the CHI side with the value that is programmed into this register.

2.11.3 QoS configuration example

This example configuration demonstrates the QoS mechanisms and their contribution to the overall QoS solution.

It is the responsibility of the SoC designer and system programmer to configure CMN-700 to meet the specific requirements of the system and expected workloads.

System operating conditions

The example QoS configuration assumes the following:

- Four processor clusters:
 - Bimodal operation. A processor cluster is latency-sensitive when bandwidth per cluster is \leq 2GB/s, and bandwidth-hungry, and therefore latency-tolerant, when bandwidth per cluster is $>$ 2GB/s.
 - 16 outstanding combined reads and writes.
 - 10GB/s maximum bandwidth per cluster.
 - 25GB/s maximum aggregate bandwidth across all processor clusters.
- Four peripheral devices with bounded latency requirements:
 - Each device is the sole device that is connected to ACE-Lite interface 0 on four different RN-I bridges.
 - 1 microsecond maximum latency requirement.
 - 4GB/s maximum bandwidth per device.
 - 210GB/s maximum aggregate bandwidth across all devices.
- 14 peripheral bandwidth-hungry devices:
 - Connected to all remaining RN-I ACE-Lite interfaces.
 - 12GB/s read or write bandwidth per device, with a combined maximum of 24GB/s.
 - 60GB/s maximum aggregate bandwidth across all devices.
- All devices can be concurrently active.
- 80GB/s maximum aggregate bandwidth across all devices.

HN-F QoS classes

For the QoS ranges and class values in HN-F, refer to [*Table 2-79 QoS classes in HN-F on page 2-211*](#).

QoS regulator settings

To meet the bandwidth and latency requirements of the described system configuration, CMN-700 QoS regulators can be configured with the settings as described in the following table.

Table 2-80 QoS regulation settings

Device	Regulation type	Regulation parameter	QoS range	QoS scale
Processor	Latency	60ns max latency	11-13	8-9
Real-time peripheral	Override (constant value)	Constant	15	n/a
High-bandwidth peripheral	Override (constant value)	Constant	8	n/a

The latency specification for real-time peripherals must be sufficiently far below the maximum real-time constraint to allow the control loop in the QoS regulator to adjust based on achieved latency, without violating the maximum latency requirement.

To meet the bandwidth and latency requirements of the described system configuration, HN-F QoS reservation values can be configured (based on 32-entry POCQ with one entry for SF back invalidations) as summarized in the following table.

Table 2-81 QoS class and reservation value settings

QoS class	QoS reservation value
HighHigh	31
High	30
Medium	15
Low	5

These settings enable the following system functionality:

- Real-time devices are QPV-15, ensuring their transactions meet their bounded latency requirements.
- The processor QPV is higher than the bandwidth-hungry devices, second only to the real-time devices, and therefore generally achieves minimum latency, except in the event of high-bandwidth real-time traffic.
- Real-time devices can be configured to have all of the HN-F POCQ buffering available to them, to prevent bandwidth limitations from impacting achieved latency.
- Real-time devices always have buffering available to them throughout the entirety of CMN-700 preventing Head-of-line blocking from lower-priority or higher-latency transactions.

Chapter 3

Programmers model

This chapter describes the application-level registers and provides an overview for programming the CMN-700 interconnect.

It contains the following sections:

- [*3.1 About the programmers model* on page 3-215](#).
- [*3.2 Register summary* on page 3-218](#).
- [*3.3 Register descriptions* on page 3-249](#).
- [*3.4 CMN-700 programming* on page 3-1238](#).
- [*3.5 CML programming* on page 3-1257](#).

3.1 About the programmers model

A CMN-700 interconnect consists of various components, such as XP, RN-I, or DTC, that are accessed through memory mapped registers for configuration, topology, and status information.

The memory mapped registers are organized in a series of 64KB regions. They are accessed through CHI read and write commands.

The memory mapped registers are organized in a series of 64KB regions. They are accessed through CHI, AXI, or APB read and write commands. APB accesses to the registers occur through the CMN-700 HN-D APB interface.

A full description of a CMN-700 interconnect consists of a list of components, the compile-time configuration options for each component, and the connectivity between the components. Software can determine the full configuration of the CMN-700 interconnect through a sequence of accesses to the configuration register space.

This section contains the following subsections:

- [3.1.1 Node configuration register address mapping on page 3-215](#).
- [3.1.2 Global configuration register region on page 3-215](#).
- [3.1.3 XP configuration register region on page 3-216](#).
- [3.1.4 Component configuration register region on page 3-216](#).
- [3.1.5 Requirements of configuration register reads and writes on page 3-216](#).

3.1.1 Node configuration register address mapping

All CMN-700 configuration registers are mapped to a specific address range that is divided into sections for individual components.

The configuration register address space starts at PERIPHBASE. For a system with X and Y dimensions of eight or less, the address space has a maximum size of 256MB. For a system with X and Y dimensions of nine or more, the address space has a maximum size of 1GB.

The **CFGM_PERIPHBASE** input signal controls the reset value of PERIPHBASE. Configuration register accesses through the HN-D APB interface use the same addressing scheme as the CHI and AXI interfaces. However, only 32 bits of the address are provided to the APB interface. Configuration register accesses through the HN-D APB interface do not include the PERIPHBASE offset and the base for these accesses is zero.

All configuration, information, and status registers in a CMN-700 interconnect are grouped into 64KB regions each associated with a CMN-700 component instance. The base address of each region can be determined at compile time, or determined at runtime through a software discovery mechanism.

Software discovery consists of three steps:

1. Read information in the 64KB region at offset **0x0**. This information determines the number of XPs in CMN-700 and the offset from PERIPHBASE for the 64KB region of each XP.
2. Read information in the 64KB region that is associated with each XP. This information determines the components that are associated with that XP, topology information for those components, and the offset from PERIPHBASE for each component 64KB region.
3. Read information in the 64KB region that is associated with the component. This information determines the type of block and the configuration details of the component.

For more information on these steps, see [2.5.4 Discovery tree structure on page 2-169](#).

With this sequence, software can build a list of all components in the system and the addresses of their respective 64KB configuration regions.

3.1.2 Global configuration register region

The 64KB block at offset **0x0** contains global information and configuration for CMN-700, and the first level of discovery information for components in the system.

Each XP Base Address register contains the offset from PERIPHBASE for a 64KB region that contains the information about one XP. The XP Base Address register also contains discovery information for components that are associated with that XP. The XP Base address refers to the relative address of the XP configuration registers. The first level of Discovery points to each por_mxp_node_info register of the XPs.

For more register information, see [3.3.12 Configuration master register descriptions](#) on page [3-765](#).

3.1.3 XP configuration register region

Each XP has a 64KB configuration register region with information about that XP and all associated components.

Refer to [3.3.18 XP register descriptions](#) on page [3-966](#) for more information.

3.1.4 Component configuration register region

Each non-XP component has a 64KB configuration register region. This region has programmable information, status, and configuration options for that component.

The contents are listed in the following table, including the number of 8B registers which fit in the space.

Table 3-1 Configuration register region values

Register sections	Relative offset	Absolute offset	Description
Discovery register section			
NODE INFO (node type, node ID)	0x0	0x0	Up to 16 registers
CHILD INFO (number of children, offset of the first child pointer register = 0x100)	0x80	0x80	Up to 16 registers
CHILD POINTER registers	0x100	0x100	Up to 256 registers
UNIT REGISTER section	0x900		Unit-specific registers
UNIT INFO	0x0	0x900	Up to 16 registers
UNIT SECURITY	0x80	0x980	Up to 16 registers
UNIT CTRL	0x100	0xA00	Up to 16 registers
UNIT QoS	0x180	0xA80	Up to 32 registers
UNIT DEBUG	0x280	0xB80	Up to 16 registers
UNIT OTHER	0x300	0xC00	Up to 128 registers
UNIT POWER	0x700	0x1000	4KB-aligned space – 512 registers
UNIT PMU	0x1700	0x2000	4KB-aligned space – 512 registers
UNIT RAS (Secure RAS registers)	0x2700	0x3000	4KB-aligned space – 512 registers
UNIT RAS (Non-secure RAS registers)	0x2800	0x3100	4KB-aligned space – 512 registers

3.1.5 Requirements of configuration register reads and writes

Reads and writes to the CMN-700 configuration registers must meet certain requirements.

A dedicated APB slave port is provided for the access of all CMN-700 configuration registers. The APB slave port has the following properties:

- APB only supports 32-bit accesses.
- **PSTRB[3:0]** must be driven to 4'hF for a write transaction.
- Secure access requires setting **PPROT[1]** to 0.

If the following requirements are not met, UNPREDICTABLE behavior can occur:

- All accesses must be of device type, either:
 - Device, Strongly Ordered.
 - nGnRE, nGnRnE.
- All accesses must have a data size of 32 bits or 64 bits.
- All accesses must be natively aligned, that is:
 - 32-bit accesses must be aligned to a 32-bit boundary.
 - 64-bit accesses must be aligned to a 64-bit boundary.
- For configuration register writes, all bits, 32 or 64, must be written, that is, all byte lanes must be valid:
 - **WRSTB** must indicate that all bytes lanes are valid if the write transaction is from an AMBA AXI or ACE-Lite interface.
 - **BE** must indicate that all byte lanes are valid if the write transaction is sent from an AMBA 5 CHI interface.
- Secure registers can only be accessed through a Secure access, that is, NS = 0. Non-secure registers can be accessed through either a Secure or Non-secure access.

For more information on error signal handling, see [2.8 Reliability, Availability, and Serviceability on page 2-175](#).

3.2 Register summary

The register summary tables list the registers in CMN-700.

This section contains the following subsections:

- [3.2.1 CCG_HA register summary on page 3-218](#).
- [3.2.2 CCG_RA register summary on page 3-219](#).
- [3.2.3 CCLA register summary on page 3-220](#).
- [3.2.4 Configuration master register summary on page 3-222](#).
- [3.2.5 CXG_HA register summary on page 3-224](#).
- [3.2.6 CXG_RA register summary on page 3-225](#).
- [3.2.7 CXLA register summary on page 3-226](#).
- [3.2.8 DN register summary on page 3-228](#).
- [3.2.9 Debug and trace register summary on page 3-229](#).
- [3.2.10 HN-F register summary on page 3-230](#).
- [3.2.11 MTU register summary on page 3-235](#).
- [3.2.12 MXP register summary on page 3-236](#).
- [3.2.13 HN-F MPAM_NS register summary on page 3-240](#).
- [3.2.14 HN-F MPAM_S register summary on page 3-242](#).
- [3.2.15 HN-I register summary on page 3-242](#).
- [3.2.16 RN-D register summary on page 3-243](#).
- [3.2.17 RN-I register summary on page 3-244](#).
- [3.2.18 RN SAM register summary on page 3-245](#).
- [3.2.19 SBSX register summary on page 3-247](#).

3.2.1 CCG_HA register summary

This section lists the CCG_HA registers used in CMN-700.

CCG_HA register summary

The following table shows the CCG_HA registers in offset order from the base memory address

Table 3-2 CCG_HA register summary

Offset	Name	Type	Description
0x0	por_ccg_ha_node_info	RO	por_ccg_ha_node_info on page 3-275
0x8	por_ccg_ha_id	RW	por_ccg_ha_id on page 3-276
0x80	por_ccg_ha_child_info	RO	por_ccg_ha_child_info on page 3-276
0xA00	por_ccg_ha_cfg_ctl	RW	por_ccg_ha_cfg_ctl on page 3-277
0xA08	por_ccg_ha_aux_ctl	RW	por_ccg_ha_aux_ctl on page 3-278
0xA10	por_ccg_ha_mpam_control_link0	RW	por_ccg_ha_mpam_control_link0 on page 3-280
0xA18	por_ccg_ha_mpam_control_link1	RW	por_ccg_ha_mpam_control_link1 on page 3-281
0xA20	por_ccg_ha_mpam_control_link2	RW	por_ccg_ha_mpam_control_link2 on page 3-283
0x980	por_ccg_ha_secure_register_groups_override	RW	por_ccg_ha_secure_register_groups_override on page 3-284
0x900	por_ccg_ha_unit_info	RO	por_ccg_ha_unit_info on page 3-285
0x908	por_ccg_ha_unit_info2	RO	por_ccg_ha_unit_info2 on page 3-287
0x1F00	por_ccg_ha_agentid_to_linkid_reg0	RW	por_ccg_ha_agentid_to_linkid_reg0 on page 3-288

Table 3-2 CCG_HA register summary (continued)

Offset	Name	Type	Description
0x1F08	por_ccg_ha_agentid_to_linkid_reg1	RW	<i>por_ccg_ha_agentid_to_linkid_reg1</i> on page 3-289
0x1F10	por_ccg_ha_agentid_to_linkid_reg2	RW	<i>por_ccg_ha_agentid_to_linkid_reg2</i> on page 3-291
0x1F18	por_ccg_ha_agentid_to_linkid_reg3	RW	<i>por_ccg_ha_agentid_to_linkid_reg3</i> on page 3-293
0x1F20	por_ccg_ha_agentid_to_linkid_reg4	RW	<i>por_ccg_ha_agentid_to_linkid_reg4</i> on page 3-295
0x1F28	por_ccg_ha_agentid_to_linkid_reg5	RW	<i>por_ccg_ha_agentid_to_linkid_reg5</i> on page 3-296
0x1F30	por_ccg_ha_agentid_to_linkid_reg6	RW	<i>por_ccg_ha_agentid_to_linkid_reg6</i> on page 3-298
0x1F38	por_ccg_ha_agentid_to_linkid_reg7	RW	<i>por_ccg_ha_agentid_to_linkid_reg7</i> on page 3-300
0x1FF8	por_ccg_ha_agentid_to_linkid_val	RW	<i>por_ccg_ha_agentid_to_linkid_val</i> on page 3-301
0x16'hC00+#[8 *[0,1,2,...,254 ,255]}]	por_ccg_ha_rnf_exp_raid_to_ldid_re g_0-255	RW	<i>por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255</i> on page 3-302
0x2000	por_ccg_ha_pmu_event_sel	RW	<i>por_ccg_ha_pmu_event_sel</i> on page 3-304
0x1C00	por_ccg_ha_cxprtcl_link0_ctl	RW	<i>por_ccg_ha_cxprtcl_link0_ctl</i> on page 3-305
0x1C08	por_ccg_ha_cxprtcl_link0_status	RO	<i>por_ccg_ha_cxprtcl_link0_status</i> on page 3-308
0x1C10	por_ccg_ha_cxprtcl_link1_ctl	RW	<i>por_ccg_ha_cxprtcl_link1_ctl</i> on page 3-309
0x1C18	por_ccg_ha_cxprtcl_link1_status	RO	<i>por_ccg_ha_cxprtcl_link1_status</i> on page 3-312
0x1C20	por_ccg_ha_cxprtcl_link2_ctl	RW	<i>por_ccg_ha_cxprtcl_link2_ctl</i> on page 3-313
0x1C28	por_ccg_ha_cxprtcl_link2_status	RO	<i>por_ccg_ha_cxprtcl_link2_status</i> on page 3-316
0x3000	por_ccg_ha_errfr	RO	<i>por_ccg_ha_errfr</i> on page 3-317
0x3008	por_ccg_ha_errctlr	RW	<i>por_ccg_ha_errctlr</i> on page 3-319
0x3010	por_ccg_ha_errstatus	W1C	<i>por_ccg_ha_errstatus</i> on page 3-320
0x3018	por_ccg_ha_erraddr	RW	<i>por_ccg_ha_erraddr</i> on page 3-322
0x3020	por_ccg_ha_errmisc	RW	<i>por_ccg_ha_errmisc</i> on page 3-323
0x3100	por_ccg_ha_errfr_NS	RO	<i>por_ccg_ha_errfr_NS</i> on page 3-325
0x3108	por_ccg_ha_errctlr_NS	RW	<i>por_ccg_ha_errctlr_NS</i> on page 3-326
0x3110	por_ccg_ha_errstatus_NS	W1C	<i>por_ccg_ha_errstatus_NS</i> on page 3-327
0x3118	por_ccg_ha_erraddr_NS	RW	<i>por_ccg_ha_erraddr_NS</i> on page 3-329
0x3120	por_ccg_ha_errmisc_NS	RW	<i>por_ccg_ha_errmisc_NS</i> on page 3-330

3.2.2 CCG_RA register summary

This section lists the CCG_RA registers used in CMN-700.

CCG_RA register summary

The following table shows the *CCG_RA* registers in offset order from the base memory address

Table 3-3 CCG_RA register summary

Offset	Name	Type	Description
0x0	por_ccg_ra_node_info	RO	<i>por_ccg_ra_node_info</i> on page 3-806
0x80	por_ccg_ra_child_info	RO	<i>por_ccg_ra_child_info</i> on page 3-807
0x980	por_ccg_ra_secure_register_groups_override	RW	<i>por_ccg_ra_secure_register_groups_override</i> on page 3-808
0x900	por_ccg_ra_unit_info	RO	<i>por_ccg_ra_unit_info</i> on page 3-809
0xA00	por_ccg_ra_cfg_ctl	RW	<i>por_ccg_ra_cfg_ctl</i> on page 3-810
0xA08	por_ccg_ra_aux_ctl	RW	<i>por_ccg_ra_aux_ctl</i> on page 3-812
0xA18	por_ccg_ra_cbusy_limit_ctl	RW	<i>por_ccg_ra_cbusy_limit_ctl</i> on page 3-815
0x16'hC00+#{8*[0,1,2,3,4,5,6,7]}	por_ccg_ra_sam_addr_region_reg_0-7	RW	<i>por_ccg_ra_sam_addr_region_reg_0-7</i> on page 3-816
0xD00	por_ccg_ra_agentid_to_linkid_val	RW	<i>por_ccg_ra_agentid_to_linkid_val</i> on page 3-817
0x16'hd10+#[0,1,2,3,4,5,6,7]*8}	por_ccg_ra_agentid_to_linkid_reg_0-7	RW	<i>por_ccg_ra_agentid_to_linkid_reg_0-7</i> on page 3-818
0x16'he00+#[0,1,2,...,8,9]*8}	por_ccg_ra_rni_ldid_to_exp_raid_reg_0-9	RW	<i>por_ccg_ra_rni_ldid_to_exp_raid_reg_0-9</i> on page 3-820
0x16'hf00+#[0,1,2,...,8,9]*8}	por_ccg_ra_rnd_ldid_to_exp_raid_reg_0-9	RW	<i>por_ccg_ra_rnd_ldid_to_exp_raid_reg_0-9</i> on page 3-821
0x16'h1000+#[0,1,2,...,126,127]*8}	por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127	RW	<i>por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127</i> on page 3-823
0x16'h1400+#[0,1,2,...,126,127]*8}	por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127	RO	<i>por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127</i> on page 3-824
0x16'h1800+#[0,1,2,...,126,127]*8}	por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127	RW	<i>por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127</i> on page 3-826
0x2000	por_ccg_ra_pmu_event_sel	RW	<i>por_ccg_ra_pmu_event_sel</i> on page 3-827
0x1C00	por_ccg_ra_ccprtcl_link0_ctl	RW	<i>por_ccg_ra_ccprtcl_link0_ctl</i> on page 3-829
0x1C08	por_ccg_ra_cxprtcl_link0_status	RO	<i>por_ccg_ra_cxpertcl_link0_status</i> on page 3-833
0x1C10	por_ccg_ra_ccprtcl_link1_ctl	RW	<i>por_ccg_ra_ccprtcl_link1_ctl</i> on page 3-834
0x1C18	por_ccg_ra_cxpertcl_link1_status	RO	<i>por_ccg_ra_cxpertcl_link1_status</i> on page 3-838
0x1C20	por_ccg_ra_ccprtcl_link2_ctl	RW	<i>por_ccg_ra_ccprtcl_link2_ctl</i> on page 3-839
0x1C28	por_ccg_ra_cxpertcl_link2_status	RO	<i>por_ccg_ra_cxpertcl_link2_status</i> on page 3-843

3.2.3 CCLA register summary

This section lists the CCLA registers used in CMN-700.

CCLA register summary

The following table shows the *CCLA* registers in offset order from the base memory address

Table 3-4 CCLA register summary

Offset	Name	Type	Description
0x0	por_ccla_node_info	RO	<i>por_ccla_node_info</i> on page 3-519
0x80	por_ccla_child_info	RO	<i>por_ccla_child_info</i> on page 3-520
0x988	por_ccla_secure_register_groups_override	RW	<i>por_ccla_secure_register_groups_override</i> on page 3-521
0x910	por_ccla_unit_info	RO	<i>por_ccla_unit_info</i> on page 3-522
0xB00	por_ccla_cfg_ctl	RW	<i>por_ccla_cfg_ctl</i> on page 3-524
0xB08	por_ccla_aux_ctl	RW	<i>por_ccla_aux_ctl</i> on page 3-525
0xC00	por_ccla_ccix_prop_capabilities	RO	<i>por_ccla_ccix_prop_capabilities</i> on page 3-529
0xC08	por_ccla_cxs_attr_capabilities	RO	<i>por_ccla_cxs_attr_capabilities</i> on page 3-530
0xD00	por_ccla_permmsg_pyld_0_63	RW	<i>por_ccla_permmsg_pyld_0_63</i> on page 3-532
0xD08	por_ccla_permmsg_pyld_64_127	RW	<i>por_ccla_permmsg_pyld_64_127</i> on page 3-533
0xD10	por_ccla_permmsg_pyld_128_191	RW	<i>por_ccla_permmsg_pyld_128_191</i> on page 3-534
0xD18	por_ccla_permmsg_pyld_192_255	RW	<i>por_ccla_permmsg_pyld_192_255</i> on page 3-535
0xD20	por_ccla_permmsg_ctl	RW	<i>por_ccla_permmsg_ctl</i> on page 3-536
0xD28	por_ccla_err_agent_id	RW	<i>por_ccla_err_agent_id</i> on page 3-537
0xD30	por_ccla_agentid_to_portid_reg0	RW	<i>por_ccla_agentid_to_portid_reg0</i> on page 3-538
0xD38	por_ccla_agentid_to_portid_reg1	RW	<i>por_ccla_agentid_to_portid_reg1</i> on page 3-539
0xD40	por_ccla_agentid_to_portid_reg2	RW	<i>por_ccla_agentid_to_portid_reg2</i> on page 3-541
0xD48	por_ccla_agentid_to_portid_reg3	RW	<i>por_ccla_agentid_to_portid_reg3</i> on page 3-543
0xD50	por_ccla_agentid_to_portid_reg4	RW	<i>por_ccla_agentid_to_portid_reg4</i> on page 3-545
0xD58	por_ccla_agentid_to_portid_reg5	RW	<i>por_ccla_agentid_to_portid_reg5</i> on page 3-546
0xD60	por_ccla_agentid_to_portid_reg6	RW	<i>por_ccla_agentid_to_portid_reg6</i> on page 3-548
0xD68	por_ccla_agentid_to_portid_reg7	RW	<i>por_ccla_agentid_to_portid_reg7</i> on page 3-550
0xD70	por_ccla_agentid_to_portid_val	RW	<i>por_ccla_agentid_to_portid_val</i> on page 3-551
0xD78	por_ccla_portfwd_ctl	RW	<i>por_ccla_portfwd_ctl</i> on page 3-552
0xD80	por_ccla_portfwd_status	RO	<i>por_ccla_portfwd_status</i> on page 3-553
0xE00	por_ccla_cxl_link_rx_credit_ctl	RW	<i>por_ccla_cxl_link_rx_credit_ctl</i> on page 3-554
0xE08	por_ccla_cxl_link_rx_credit_return_stat	RO	<i>por_ccla_cxl_link_rx_credit_return_stat</i> on page 3-555
0xE10	por_ccla_cxl_link_tx_credit_return_stat	RO	<i>por_ccla_cxl_link_tx_credit_return_stat</i> on page 3-556
0xE18	por_ccla_cxl_link_layer_defeature	RW	<i>por_ccla_cxl_link_layer_defeature</i> on page 3-558
0xE20	por_ccla_ull_ctl	RW	<i>por_ccla_ull_ctl</i> on page 3-558

Table 3-4 CCLA register summary (continued)

Offset	Name	Type	Description
0xE28	por_ccla_ull_status	RO	<i>por_ccla_ull_status</i> on page 3-560
0x2008	por_ccla_pmu_event_sel	RW	<i>por_ccla_pmu_event_sel</i> on page 3-560
0x980	por_ccla_rni_secure_register_groups_override	RW	<i>por_ccla_rni_secure_register_groups_override</i> on page 3-562
0x900	por_ccla_rni_unit_info	RO	<i>por_ccla_rni_unit_info</i> on page 3-563
0x908	por_ccla_rni_unit_info2	RO	<i>por_ccla_rni_unit_info2</i> on page 3-565
0xA00	por_ccla_rni_cfg_ctl	RW	<i>por_ccla_rni_cfg_ctl</i> on page 3-566
0xA08	por_ccla_rni_aux_ctl	RW	<i>por_ccla_rni_aux_ctl</i> on page 3-568
0x16'hA10+#[0, 1, 2]*8}	por_ccla_rni_s_0-2_port_control	RW	<i>por_ccla_rni_s_0-2_port_control</i> on page 3-571
0x16'hA28+#[0, 1, 2]*8}	por_ccla_rni_s_0-2_mpam_control	RW	<i>por_ccla_rni_s_0-2_mpam_control</i> on page 3-573
0x16'hA80+#[0, 1, 2]*32}	por_ccla_rni_s_0-2_qos_control	RW	<i>por_ccla_rni_s_0-2_qos_control</i> on page 3-574
0x16'hA88+#[0, 1, 2]*32}	por_ccla_rni_s_0-2_qos_lat_tgt	RW	<i>por_ccla_rni_s_0-2_qos_lat_tgt</i> on page 3-576
0x16'hA90+#[0, 1, 2]*32}	por_ccla_rni_s_0-2_qos_lat_scale	RW	<i>por_ccla_rni_s_0-2_qos_lat_scale</i> on page 3-577
0x16'hA98+#[0, 1, 2]*32}	por_ccla_rni_s_0-2_qos_lat_range	RW	<i>por_ccla_rni_s_0-2_qos_lat_range</i> on page 3-579
0x2000	por_ccla_rni_pmu_event_sel	RW	<i>por_ccla_rni_pmu_event_sel</i> on page 3-580

3.2.4 Configuration master register summary

This section lists the configuration master registers used in CMN-700.

CFGM register summary

The following table shows the *CFGM* registers in offset order from the base memory address

Table 3-5 CFGM register summary

Offset	Name	Type	Description
0x0	por_cfgm_node_info	RO	<i>por_cfgm_node_info</i> on page 3-765
0x8	por_cfgm_periph_id_0_periph_id_1	RO	<i>por_cfgm_periph_id_0_periph_id_1</i> on page 3-766
0x10	por_cfgm_periph_id_2_periph_id_3	RO	<i>por_cfgm_periph_id_2_periph_id_3</i> on page 3-767
0x18	por_cfgm_periph_id_4_periph_id_5	RO	<i>por_cfgm_periph_id_4_periph_id_5</i> on page 3-768
0x20	por_cfgm_periph_id_6_periph_id_7	RO	<i>por_cfgm_periph_id_6_periph_id_7</i> on page 3-769
0x28	por_cfgm_component_id_0_component_id_1	RO	<i>por_cfgm_component_id_0_component_id_1</i> on page 3-770

Table 3-5 CFGM register summary (continued)

Offset	Name	Type	Description
0x30	por_cfgm_component_id_2_component_id_3	RO	<i>por_cfgm_component_id_2_component_id_3</i> on page 3-771
0x80	por_cfgm_child_info	RO	<i>por_cfgm_child_info</i> on page 3-772
0x980	por_cfgm_secure_access	RW	<i>por_cfgm_secure_access</i> on page 3-773
0x988	por_cfgm_secure_register_groups_override	RW	<i>por_cfgm_secure_register_groups_override</i> on page 3-774
0x16'h3000+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_mxp_0-7	RO	<i>por_cfgm_errgsr_mxp_0-7</i> on page 3-775
0x16'h3040+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_mxp_0-7_NS	RO	<i>por_cfgm_errgsr_mxp_0-7_NS</i> on page 3-776
0x16'h3080+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_hni_0-7	RO	<i>por_cfgm_errgsr_hni_0-7</i> on page 3-777
0x16'h30C0+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_hni_0-7_NS	RO	<i>por_cfgm_errgsr_hni_0-7_NS</i> on page 3-778
0x16'h3100+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_hnf_0-7	RO	<i>por_cfgm_errgsr_hnf_0-7</i> on page 3-779
0x16'h3140+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_hnf_0-7_NS	RO	<i>por_cfgm_errgsr_hnf_0-7_NS</i> on page 3-780
0x16'h3180+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_sbsx_0-7	RO	<i>por_cfgm_errgsr_sbsx_0-7</i> on page 3-780
0x16'h31C0+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_sbsx_0-7_NS	RO	<i>por_cfgm_errgsr_sbsx_0-7_NS</i> on page 3-781
0x16'h3200+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_cxg_0-7	RO	<i>por_cfgm_errgsr_cxg_0-7</i> on page 3-782
0x16'h3240+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_cxg_0-7_NS	RO	<i>por_cfgm_errgsr_cxg_0-7_NS</i> on page 3-783
0x16'h3280+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_mtsx_0-7	RO	<i>por_cfgm_errgsr_mtsx_0-7</i> on page 3-784
0x16'h32C0+#{8*[0,1,2,3,4,5,6,7]}	por_cfgm_errgsr_mtsx_0-7_NS	RO	<i>por_cfgm_errgsr_mtsx_0-7_NS</i> on page 3-785
0x3FA8	por_cfgm_errdevaff	RO	<i>por_cfgm_errdevaff</i> on page 3-786
0x3FB8	por_cfgm_errdevarch	RO	<i>por_cfgm_errdevarch</i> on page 3-786
0x3FC8	por_cfgm_erridr	RO	<i>por_cfgm_erridr</i> on page 3-787
0x3FD0	por_cfgm_errpidr45	RO	<i>por_cfgm_errpidr45</i> on page 3-788
0x3FD8	por_cfgm_errpidr67	RO	<i>por_cfgm_errpidr67</i> on page 3-789
0x3FE0	por_cfgm_errpidr01	RO	<i>por_cfgm_errpidr01</i> on page 3-790
0x3FE8	por_cfgm_errpidr23	RO	<i>por_cfgm_errpidr23</i> on page 3-791
0x3FF0	por_cfgm_errcidr01	RO	<i>por_cfgm_errcidr01</i> on page 3-792
0x3FF8	por_cfgm_errcidr23	RO	<i>por_cfgm_errcidr23</i> on page 3-793
0x900	por_info_global	RO	<i>por_info_global</i> on page 3-794
0x908	por_info_global_1	RO	<i>por_info_global_1</i> on page 3-796
0x1C00	por_ppu_int_enable	RW	<i>por_ppu_int_enable</i> on page 3-797
0x1C08	por_ppu_int_enable_1	RW	<i>por_ppu_int_enable_1</i> on page 3-798
0x1C10	por_ppu_int_status	W1C	<i>por_ppu_int_status</i> on page 3-799
0x1C18	por_ppu_int_status_1	W1C	<i>por_ppu_int_status_1</i> on page 3-800

Table 3-5 CFGM register summary (continued)

Offset	Name	Type	Description
0x1C20	por_ppu_qactive_hyst	RW	por_ppu_qactive_hyst on page 3-801
0x1C28	por_mpam_s_err_int_status	W1C	por_mpam_s_err_int_status on page 3-802
0x1C30	por_mpam_s_err_int_status_1	W1C	por_mpam_s_err_int_status_1 on page 3-802
0x1C38	por_mpam_ns_err_int_status	W1C	por_mpam_ns_err_int_status on page 3-803
0x1C40	por_mpam_ns_err_int_status_1	W1C	por_mpam_ns_err_int_status_1 on page 3-804
0x16'h100+#{8*[0,1,2,...,254,255]}	por_cfgm_child_pointer_0-255	RO	por_cfgm_child_pointer_0-255 on page 3-805

3.2.5 CXG_HA register summary

This section lists the CXG_HA registers used in CMN-700.

CXG_HA register summary

The following table shows the *CXG_HA* registers in offset order from the base memory address

Table 3-6 CXG_HA register summary

Offset	Name	Type	Description
0x0	por_cxg_ha_node_info	RO	por_cxg_ha_node_info on page 3-584
0x8	por_cxg_ha_id	RW	por_cxg_ha_id on page 3-585
0x80	por_cxg_ha_child_info	RO	por_cxg_ha_child_info on page 3-585
0xA00	por_cxg_ha_cfg_ctl	RW	por_cxg_ha_cfg_ctl on page 3-586
0xA08	por_cxg_ha_aux_ctl	RW	por_cxg_ha_aux_ctl on page 3-587
0xA10	por_cxg_ha_mpam_control	RW	por_cxg_ha_mpam_control on page 3-589
0x980	por_cxg_ha_secure_register_groups_override	RW	por_cxg_ha_secure_register_groups_override on page 3-590
0x900	por_cxg_ha_unit_info	RO	por_cxg_ha_unit_info on page 3-591
0x908	por_cxg_ha_unit_info2	RO	por_cxg_ha_unit_info2 on page 3-593
0x1F00	por_cxg_ha_agentid_to_linkid_reg0	RW	por_cxg_ha_agentid_to_linkid_reg0 on page 3-594
0x1F08	por_cxg_ha_agentid_to_linkid_reg1	RW	por_cxg_ha_agentid_to_linkid_reg1 on page 3-595
0x1F10	por_cxg_ha_agentid_to_linkid_reg2	RW	por_cxg_ha_agentid_to_linkid_reg2 on page 3-597
0x1F18	por_cxg_ha_agentid_to_linkid_reg3	RW	por_cxg_ha_agentid_to_linkid_reg3 on page 3-599
0x1F20	por_cxg_ha_agentid_to_linkid_reg4	RW	por_cxg_ha_agentid_to_linkid_reg4 on page 3-601
0x1F28	por_cxg_ha_agentid_to_linkid_reg5	RW	por_cxg_ha_agentid_to_linkid_reg5 on page 3-602

Table 3-6 CXG_HA register summary (continued)

Offset	Name	Type	Description
0x1F30	por_cxg_ha_agentid_to_linkid_reg6	RW	por_cxg_ha_agentid_to_linkid_reg6 on page 3-604
0x1F38	por_cxg_ha_agentid_to_linkid_reg7	RW	por_cxg_ha_agentid_to_linkid_reg7 on page 3-606
0xFF8	por_cxg_ha_agentid_to_linkid_val	RW	por_cxg_ha_agentid_to_linkid_val on page 3-607
0x16'hC00+#{8 *[0,1,2,...,254 ,255]}	por_cxg_ha_rnf_exp_raid_to_ldid_re g_0-255	RW	por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255 on page 3-608
0x2000	por_cxg_ha_pmu_event_sel	RW	por_cxg_ha_pmu_event_sel on page 3-610
0x1C00	por_cxg_ha_cxpctl_link0_ctl	RW	por_cxg_ha_cxpctl_link0_ctl on page 3-611
0x1C08	por_cxg_ha_cxpctl_link0_status	RO	por_cxg_ha_cxpctl_link0_status on page 3-614
0x1C10	por_cxg_ha_cxpctl_link1_ctl	RW	por_cxg_ha_cxpctl_link1_ctl on page 3-615
0x1C18	por_cxg_ha_cxpctl_link1_status	RO	por_cxg_ha_cxpctl_link1_status on page 3-617
0x1C20	por_cxg_ha_cxpctl_link2_ctl	RW	por_cxg_ha_cxpctl_link2_ctl on page 3-618
0x1C28	por_cxg_ha_cxpctl_link2_status	RO	por_cxg_ha_cxpctl_link2_status on page 3-620
0x3000	por_cxg_ha_errfr	RO	por_cxg_ha_errfr on page 3-622
0x3008	por_cxg_ha_errctlr	RW	por_cxg_ha_errctlr on page 3-623
0x3010	por_cxg_ha_errstatus	W1C	por_cxg_ha_errstatus on page 3-624
0x3018	por_cxg_ha_erraddr	RW	por_cxg_ha_erraddr on page 3-626
0x3020	por_cxg_ha_errmisc	RW	por_cxg_ha_errmisc on page 3-628
0x3100	por_cxg_ha_errfr_NS	RO	por_cxg_ha_errfr_NS on page 3-629
0x3108	por_cxg_ha_errctlr_NS	RW	por_cxg_ha_errctlr_NS on page 3-630
0x3110	por_cxg_ha_errstatus_NS	W1C	por_cxg_ha_errstatus_NS on page 3-631
0x3118	por_cxg_ha_erraddr_NS	RW	por_cxg_ha_erraddr_NS on page 3-633
0x3120	por_cxg_ha_errmisc_NS	RW	por_cxg_ha_errmisc_NS on page 3-634

3.2.6 CXG_RA register summary

This section lists the CXG_RA registers used in CMN-700.

CXG_RA register summary

The following table shows the CXG_RA registers in offset order from the base memory address

Table 3-7 CXG_RA register summary

Offset	Name	Type	Description
0x0	por_cxg_ra_node_info	RO	por_cxg_ra_node_info on page 3-331
0x80	por_cxg_ra_child_info	RO	por_cxg_ra_child_info on page 3-332
0x980	por_cxg_ra_secure_register_groups_override	RW	por_cxg_ra_secure_register_groups_override on page 3-333

Table 3-7 CXG_RA register summary (continued)

Offset	Name	Type	Description
0x900	por_cxg_ra_unit_info	RO	por_cxg_ra_unit_info on page 3-334
0xA00	por_cxg_ra_cfg_ctl	RW	por_cxg_ra_cfg_ctl on page 3-336
0xA08	por_cxg_ra_aux_ctl	RW	por_cxg_ra_aux_ctl on page 3-338
0xA18	por_cxg_ra_cbusy_limit_ctl	RW	por_cxg_ra_cbusy_limit_ctl on page 3-340
0x16'hC00+#{8*[0,1,2,3,4,5,6,7]}	por_cxg_ra_sam_addr_region_reg_0-7	RW	por_cxg_ra_sam_addr_region_reg_0-7 on page 3-341
0xD00	por_cxg_ra_agentid_to_linkid_val	RW	por_cxg_ra_agentid_to_linkid_val on page 3-342
0x16'hD10+#[0,1,2,3,4,5,6,7]*8}	por_cxg_ra_agentid_to_linkid_reg_0-7	RW	por_cxg_ra_agentid_to_linkid_reg_0-7 on page 3-343
0x16'hE00+#[0,1,2,...,8,9]*8}	por_cxg_ra_rni_ldid_to_exp_raid_reg_0-9	RW	por_cxg_ra_rni_ldid_to_exp_raid_reg_0-9 on page 3-345
0x16'hF00+#[0,1,2,...,8,9]*8}	por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9	RW	por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9 on page 3-346
0x16'h1000+#[0,1,2,...,126,127]*8}	por_cxg_ra_rnf_ldid_to_exp_raid_reg_0-127	RW	por_cxg_ra_rnf_ldid_to_exp_raid_reg_0-127 on page 3-348
0x16'h1400+#[0,1,2,...,126,127]*8}	por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127	RO	por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127 on page 3-349
0x16'h1800+#[0,1,2,...,126,127]*8}	por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127	RW	por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 on page 3-351
0x2000	por_cxg_ra_pmu_event_sel	RW	por_cxg_ra_pmu_event_sel on page 3-352
0x1C00	por_cxg_ra_cxrptcl_link0_ctl	RW	por_cxg_ra_cxrptcl_link0_ctl on page 3-354
0x1C08	por_cxg_ra_cxrptcl_link0_status	RO	por_cxg_ra_cxrptcl_link0_status on page 3-357
0x1C10	por_cxg_ra_cxrptcl_link1_ctl	RW	por_cxg_ra_cxrptcl_link1_ctl on page 3-358
0x1C18	por_cxg_ra_cxrptcl_link1_status	RO	por_cxg_ra_cxrptcl_link1_status on page 3-361
0x1C20	por_cxg_ra_cxrptcl_link2_ctl	RW	por_cxg_ra_cxrptcl_link2_ctl on page 3-362
0x1C28	por_cxg_ra_cxrptcl_link2_status	RO	por_cxg_ra_cxrptcl_link2_status on page 3-365

3.2.7 CXLA register summary

This section lists the CXLA registers used in CMN-700.

CXLA register summary

The following table shows the CXLA registers in offset order from the base memory address

Table 3-8 CXLA register summary

Offset	Name	Type	Description
0x0	por_cxla_node_info	RO	<i>por_cxla_node_info</i> on page 3-438
0x80	por_cxla_child_info	RO	<i>por_cxla_child_info</i> on page 3-439
0x980	por_cxla_secure_register_groups_override	RW	<i>por_cxla_secure_register_groups_override</i> on page 3-440
0x900	por_cxla_unit_info	RO	<i>por_cxla_unit_info</i> on page 3-441
0xA00	por_cxla_cfg_ctl	RW	<i>por_cxla_cfg_ctl</i> on page 3-443
0xA08	por_cxla_aux_ctl	RW	<i>por_cxla_aux_ctl</i> on page 3-444
0xC00	por_cxla_ccix_prop_capabilities	RO	<i>por_cxla_ccix_prop_capabilities</i> on page 3-448
0xC08	por_cxla_ccix_prop_configured	RW	<i>por_cxla_ccix_prop_configured</i> on page 3-450
0xC10	por_cxla_tx_cxs_attr_capabilities	RO	<i>por_cxla_tx_cxs_attr_capabilities</i> on page 3-452
0xC18	por_cxla_rx_cxs_attr_capabilities	RO	<i>por_cxla_rx_cxs_attr_capabilities</i> on page 3-454
0xC30	por_cxla_agentid_to_linkid_reg0	RW	<i>por_cxla_agentid_to_linkid_reg0</i> on page 3-455
0xC38	por_cxla_agentid_to_linkid_reg1	RW	<i>por_cxla_agentid_to_linkid_reg1</i> on page 3-457
0xC40	por_cxla_agentid_to_linkid_reg2	RW	<i>por_cxla_agentid_to_linkid_reg2</i> on page 3-459
0xC48	por_cxla_agentid_to_linkid_reg3	RW	<i>por_cxla_agentid_to_linkid_reg3</i> on page 3-460
0xC50	por_cxla_agentid_to_linkid_reg4	RW	<i>por_cxla_agentid_to_linkid_reg4</i> on page 3-462
0xC58	por_cxla_agentid_to_linkid_reg5	RW	<i>por_cxla_agentid_to_linkid_reg5</i> on page 3-464
0xC60	por_cxla_agentid_to_linkid_reg6	RW	<i>por_cxla_agentid_to_linkid_reg6</i> on page 3-465
0xC68	por_cxla_agentid_to_linkid_reg7	RW	<i>por_cxla_agentid_to_linkid_reg7</i> on page 3-467
0xC70	por_cxla_agentid_to_linkid_val	RW	<i>por_cxla_agentid_to_linkid_val</i> on page 3-469
0xC78	por_cxla_linkid_to_pcie_bus_num	RW	<i>por_cxla_linkid_to_pcie_bus_num</i> on page 3-469
0xC80	por_cxla_tlp_hdr_fields	RW	<i>por_cxla_tlp_hdr_fields</i> on page 3-471
0xD00	por_cxla_permmsg_pyld_0_63	RW	<i>por_cxla_permmsg_pyld_0_63</i> on page 3-472
0xD08	por_cxla_permmsg_pyld_64_127	RW	<i>por_cxla_permmsg_pyld_64_127</i> on page 3-473
0xD10	por_cxla_permmsg_pyld_128_191	RW	<i>por_cxla_permmsg_pyld_128_191</i> on page 3-474
0xD18	por_cxla_permmsg_pyld_192_255	RW	<i>por_cxla_permmsg_pyld_192_255</i> on page 3-475
0xD20	por_cxla_permmsg_ctl	RW	<i>por_cxla_permmsg_ctl</i> on page 3-476
0xD28	por_cxla_err_agent_id	RW	<i>por_cxla_err_agent_id</i> on page 3-477

Table 3-8 CXLA register summary (continued)

Offset	Name	Type	Description
0xD30	por_cxla_agentid_to_portid_reg0	RW	<i>por_cxla_agentid_to_portid_reg0</i> on page 3-478
0xD38	por_cxla_agentid_to_portid_reg1	RW	<i>por_cxla_agentid_to_portid_reg1</i> on page 3-479
0xD40	por_cxla_agentid_to_portid_reg2	RW	<i>por_cxla_agentid_to_portid_reg2</i> on page 3-481
0xD48	por_cxla_agentid_to_portid_reg3	RW	<i>por_cxla_agentid_to_portid_reg3</i> on page 3-483
0xD50	por_cxla_agentid_to_portid_reg4	RW	<i>por_cxla_agentid_to_portid_reg4</i> on page 3-485
0xD58	por_cxla_agentid_to_portid_reg5	RW	<i>por_cxla_agentid_to_portid_reg5</i> on page 3-486
0xD60	por_cxla_agentid_to_portid_reg6	RW	<i>por_cxla_agentid_to_portid_reg6</i> on page 3-488
0xD68	por_cxla_agentid_to_portid_reg7	RW	<i>por_cxla_agentid_to_portid_reg7</i> on page 3-490
0xD70	por_cxla_agentid_to_portid_val	RW	<i>por_cxla_agentid_to_portid_val</i> on page 3-491
0xD78	por_cxla_portfwd_ctl	RW	<i>por_cxla_portfwd_ctl</i> on page 3-492
0xD80	por_cxla_portfwd_status	RO	<i>por_cxla_portfwd_status</i> on page 3-493
0x2000	por_cxla_pmu_event_sel	RW	<i>por_cxla_pmu_event_sel</i> on page 3-494
0x2210	por_cxla_pmu_config	RW	<i>por_cxla_pmu_config</i> on page 3-498
0x2220	por_cxla_pmevcnt	RW	<i>por_cxla_pmevcnt</i> on page 3-499
0x2240	por_cxla_pmevcntsr	RW	<i>por_cxla_pmevcntsr</i> on page 3-500

3.2.8 DN register summary

This section lists the DN registers used in CMN-700.

DN register summary

The following table shows the *DN* registers in offset order from the base memory address

Table 3-9 DN register summary

Offset	Name	Type	Description
0x0	por_dn_node_info	RO	<i>por_dn_node_info</i> on page 3-501
0x80	por_dn_child_info	RO	<i>por_dn_child_info</i> on page 3-502
0x900	por_dn_build_info	RO	<i>por_dn_build_info</i> on page 3-503
0x980	por_dn_secure_register_groups_override	RW	<i>por_dn_secure_register_groups_override</i> on page 3-504
0xA00	por_dn_cfg_ctl	RW	<i>por_dn_cfg_ctl</i> on page 3-505
0xA08	por_dn_aux_ctl	RW	<i>por_dn_aux_ctl</i> on page 3-506

Table 3-9 DN register summary (continued)

Offset	Name	Type	Description
0x16'hC00+#{56*[0,1,2,...,14,15]}	por_dn_vmf_0-15_ctrl	RW	por_dn_vmf_0-15_ctrl on page 3-507
0x16'hC00+#{56*[0,1,2,...,14,15]+8}	por_dn_vmf_0-15_rnf0	RW	por_dn_vmf_0-15_rnf0 on page 3-509
0x16'hC00+#{56*[0,1,2,...,14,15]+16}	por_dn_vmf_0-15_rnf1	RW	por_dn_vmf_0-15_rnf1 on page 3-510
0x16'hC00+#{56*[0,1,2,...,14,15]+24}	por_dn_vmf_0-15_rnf2	RW	por_dn_vmf_0-15_rnf2 on page 3-511
0x16'hC00+#{56*[0,1,2,...,14,15]+32}	por_dn_vmf_0-15_rnf3	RW	por_dn_vmf_0-15_rnf3 on page 3-512
0x16'hC00+#{56*[0,1,2,...,14,15]+40}	por_dn_vmf_0-15_rnd	RW	por_dn_vmf_0-15_rnd on page 3-513
0x16'hC00+#{56*[0,1,2,...,14,15]+48}	por_dn_vmf_0-15_cxra	RW	por_dn_vmf_0-15_cxra on page 3-513
0x16'hF80+#{8*[0,1,2,3]}	por_dn_domain_rnf_0-3	RW	por_dn_domain_rnf_0-3 on page 3-514
0xFA0	por_dn_domain_rnd	RW	por_dn_domain_rnd on page 3-515
0xFA8	por_dn_domain_cxra	RW	por_dn_domain_cxra on page 3-516
0x2000	por_dn_pmu_event_sel	RW	por_dn_pmu_event_sel on page 3-517

3.2.9 Debug and trace register summary

This section lists the debug and trace registers used in CMN-700.

DT register summary

The following table shows the *DT* registers in offset order from the base memory address

Table 3-10 DT register summary

Offset	Name	Type	Description
0x0	por_dt_node_info	RO	por_dt_node_info on page 3-844
0x80	por_dt_child_info	RO	por_dt_child_info on page 3-845
0x980	por_dt_secure_access	RW	por_dt_secure_access on page 3-846
0xA00	por_dt_dtc_ctl	RW	por_dt_dtc_ctl on page 3-847
0xA10	por_dt_trigger_status	RO	por_dt_trigger_status on page 3-849
0xA20	por_dt_trigger_status_clr	WO	por_dt_trigger_status_clr on page 3-850
0xA30	por_dt_trace_control	RW	por_dt_trace_control on page 3-851

Table 3-10 DT register summary (continued)

Offset	Name	Type	Description
0xA48	por_dt_traceid	RW	<i>por_dt_traceid</i> on page 3-852
0x2000	por_dt_pmevcntAB	RW	<i>por_dt_pmevcntAB</i> on page 3-853
0x2010	por_dt_pmevcntCD	RW	<i>por_dt_pmevcntCD</i> on page 3-854
0x2020	por_dt_pmevcntEF	RW	<i>por_dt_pmevcntEF</i> on page 3-854
0x2030	por_dt_pmevcntGH	RW	<i>por_dt_pmevcntGH</i> on page 3-855
0x2040	por_dt_pmccntr	RW	<i>por_dt_pmccntr</i> on page 3-856
0x2050	por_dt_pmevcntsAB	RW	<i>por_dt_pmevcntsAB</i> on page 3-857
0x2060	por_dt_pmevcntsCD	RW	<i>por_dt_pmevcntsCD</i> on page 3-858
0x2070	por_dt_pmevcntsEF	RW	<i>por_dt_pmevcntsEF</i> on page 3-859
0x2080	por_dt_pmevcntsGH	RW	<i>por_dt_pmevcntsGH</i> on page 3-859
0x2090	por_dt_pmccntrs	RW	<i>por_dt_pmccntrs</i> on page 3-860
0x2100	por_dt_pmcr	RW	<i>por_dt_pmcr</i> on page 3-861
0x2118	por_dt_pmovsr	RO	<i>por_dt_pmovsr</i> on page 3-862
0x2120	por_dt_pmovsr_clr	WO	<i>por_dt_pmovsr_clr</i> on page 3-863
0x2128	por_dt_pmssr	RO	<i>por_dt_pmssr</i> on page 3-864
0x2130	por_dt_pmsrr	WO	<i>por_dt_pmsrr</i> on page 3-865
0xFA0	por_dt_claim	RW	<i>por_dt_claim</i> on page 3-866
0xFA8	por_dt_devaff	RO	<i>por_dt_devaff</i> on page 3-867
0xFB0	por_dt_lsr	RO	<i>por_dt_lsr</i> on page 3-868
0xFB8	por_dt_authstatus_devarch	RO	<i>por_dt_authstatus_devarch</i> on page 3-869
0xFC0	por_dt_devid	RO	<i>por_dt_devid</i> on page 3-870
0xFC8	por_dt_devtype	RO	<i>por_dt_devtype</i> on page 3-871
0xFD0	por_dt_pidr45	RO	<i>por_dt_pidr45</i> on page 3-872
0xFD8	por_dt_pidr67	RO	<i>por_dt_pidr67</i> on page 3-873
0xFE0	por_dt_pidr01	RO	<i>por_dt_pidr01</i> on page 3-874
0xFE8	por_dt_pidr23	RO	<i>por_dt_pidr23</i> on page 3-875
0xFF0	por_dt_cidr01	RO	<i>por_dt_cidr01</i> on page 3-876
0xFF8	por_dt_cidr23	RO	<i>por_dt_cidr23</i> on page 3-877

3.2.10 HN-F register summary

This section lists the HN-F registers used in CMN-700.

HNF register summary

The following table shows the *HNF* registers in offset order from the base memory address

Table 3-11 HNF register summary

Offset	Name	Type	Description
0x0	por_hnf_node_info	RO	por_hnf_node_info on page 3-1058
0x80	por_hnf_child_info	RO	por_hnf_child_info on page 3-1059
0x980	por_hnf_secure_register_groups_override	RW	por_hnf_secure_register_groups_override on page 3-1060
0x900	por_hnf_unit_info	RO	por_hnf_unit_info on page 3-1061
0x908	por_hnf_unit_info_1	RO	por_hnf_unit_info_1 on page 3-1064
0xA00	por_hnf_cfg_ctl	RW	por_hnf_cfg_ctl on page 3-1066
0xA08	por_hnf_aux_ctl	RW	por_hnf_aux_ctl on page 3-1069
0xA10	por_hnf_r2_aux_ctl	RW	por_hnf_r2_aux_ctl on page 3-1073
0xA18	por_hnf_cbusy_limit_ctl	RW	por_hnf_cbusy_limit_ctl on page 3-1076
0x1C00	por_hnf_ppu_pwpr	RW	por_hnf_ppu_pwpr on page 3-1077
0x1C08	por_hnf_ppu_pwsr	RO	por_hnf_ppu_pwsr on page 3-1078
0x1C14	por_hnf_ppu_misr	RO	por_hnf_ppu_misr on page 3-1079
0x2BB0	por_hnf_ppu_idr0	RO	por_hnf_ppu_idr0 on page 3-1079
0x2BB4	por_hnf_ppu_idr1	RO	por_hnf_ppu_idr1 on page 3-1081
0x2BC8	por_hnf_ppu_iidr	RO	por_hnf_ppu_iidr on page 3-1082
0x2BCC	por_hnf_ppu_aidr	RO	por_hnf_ppu_aidr on page 3-1083
0x1D00	por_hnf_ppu_dyn_ret_threshold	RW	por_hnf_ppu_dyn_ret_threshold on page 3-1083
0xA80	por_hnf_qos_band	RO	por_hnf_qos_band on page 3-1084
0xA88	por_hnf_qos_reservation	RW	por_hnf_qos_reservation on page 3-1085
0x3000	por_hnf_errfr	RO	por_hnf_errfr on page 3-1086
0x3008	por_hnf_errctlr	RW	por_hnf_errctlr on page 3-1088
0x3010	por_hnf_errstatus	W1C	por_hnf_errstatus on page 3-1089
0x3018	por_hnf_erraddr	RW	por_hnf_erraddr on page 3-1091
0x3020	por_hnf_errmisc	RW	por_hnf_errmisc on page 3-1092
0x3030	por_hnf_err_inj	RW	por_hnf_err_inj on page 3-1094
0x3038	por_hnf_byte_par_err_inj	WO	por_hnf_byte_par_err_inj on page 3-1095
0x3100	por_hnf_errfr_NS	RO	por_hnf_errfr_NS on page 3-1096
0x3108	por_hnf_errctlr_NS	RW	por_hnf_errctlr_NS on page 3-1098
0x3110	por_hnf_errstatus_NS	W1C	por_hnf_errstatus_NS on page 3-1099
0x3118	por_hnf_erraddr_NS	RW	por_hnf_erraddr_NS on page 3-1101
0x3120	por_hnf_errmisc_NS	RW	por_hnf_errmisc_NS on page 3-1102
0xC00	por_hnf_slc_lock_ways	RW	por_hnf_slc_lock_ways on page 3-1104
0xC08	por_hnf_slc_lock_base0	RW	por_hnf_slc_lock_base0 on page 3-1105

Table 3-11 HNF register summary (continued)

Offset	Name	Type	Description
0xC10	por_hnf_slc_lock_base1	RW	por_hnf_slc_lock_base1 on page 3-1106
0xC18	por_hnf_slc_lock_base2	RW	por_hnf_slc_lock_base2 on page 3-1107
0xC20	por_hnf_slc_lock_base3	RW	por_hnf_slc_lock_base3 on page 3-1108
0xC28	por_hnf_rni_region_vec	RW	por_hnf_rni_region_vec on page 3-1109
0xC30	por_hnf_rnd_region_vec	RW	por_hnf_rnd_region_vec on page 3-1110
0xC38	por_hnf_rnf_region_vec	RW	por_hnf_rnf_region_vec on page 3-1111
0xC40	por_hnf_rnf_region_vec1	RW	por_hnf_rnf_region_vec1 on page 3-1112
0xC48	por_hnf_slcway_partition0_rnf_vec	RW	por_hnf_slcway_partition0_rnf_vec on page 3-1113
0xC50	por_hnf_slcway_partition1_rnf_vec	RW	por_hnf_slcway_partition1_rnf_vec on page 3-1114
0xC58	por_hnf_slcway_partition2_rnf_vec	RW	por_hnf_slcway_partition2_rnf_vec on page 3-1115
0xC60	por_hnf_slcway_partition3_rnf_vec	RW	por_hnf_slcway_partition3_rnf_vec on page 3-1116
0xCB0	por_hnf_slcway_partition0_rnf_vec1	RW	por_hnf_slcway_partition0_rnf_vec1 on page 3-1116
0xCB8	por_hnf_slcway_partition1_rnf_vec1	RW	por_hnf_slcway_partition1_rnf_vec1 on page 3-1117
0xCC0	por_hnf_slcway_partition2_rnf_vec1	RW	por_hnf_slcway_partition2_rnf_vec1 on page 3-1118
0xCC8	por_hnf_slcway_partition3_rnf_vec1	RW	por_hnf_slcway_partition3_rnf_vec1 on page 3-1119
0xC68	por_hnf_slcway_partition0_rni_vec	RW	por_hnf_slcway_partition0_rni_vec on page 3-1120
0xC70	por_hnf_slcway_partition1_rni_vec	RW	por_hnf_slcway_partition1_rni_vec on page 3-1121
0xC78	por_hnf_slcway_partition2_rni_vec	RW	por_hnf_slcway_partition2_rni_vec on page 3-1122
0xC80	por_hnf_slcway_partition3_rni_vec	RW	por_hnf_slcway_partition3_rni_vec on page 3-1122
0xC88	por_hnf_slcway_partition0_rnd_vec	RW	por_hnf_slcway_partition0_rnd_vec on page 3-1123
0xC90	por_hnf_slcway_partition1_rnd_vec	RW	por_hnf_slcway_partition1_rnd_vec on page 3-1124
0xC98	por_hnf_slcway_partition2_rnd_vec	RW	por_hnf_slcway_partition2_rnd_vec on page 3-1125
0xCA0	por_hnf_slcway_partition3_rnd_vec	RW	por_hnf_slcway_partition3_rnd_vec on page 3-1126
0xCA8	por_hnf_rn_region_lock	RW	por_hnf_rn_region_lock on page 3-1127

Table 3-11 HNF register summary (continued)

Offset	Name	Type	Description
0xCD0	por_hnf_sf_cxg_blocked_ways	RW	<i>por_hnf_sf_cxg_blocked_ways</i> on page 3-1128
0xCE0	por_hnf_cxg_ha_metadata_exclusion_list	RW	<i>por_hnf_cxg_ha_metadata_exclusion_list</i> on page 3-1129
0xCD8	por_hnf_cxg_ha_smp_exclusion_list	RW	<i>por_hnf_cxg_ha_smp_exclusion_list</i> on page 3-1130
0xCF0	hn_sam_hash_addr_mask_reg	RW	<i>hn_sam_hash_addr_mask_reg</i> on page 3-1131
0xCF8	hn_sam_region_cmp_addr_mask_reg	RW	<i>hn_sam_region_cmp_addr_mask_reg</i> on page 3-1132
0xD00	por_hnf_sam_control	RW	<i>por_hnf_sam_control</i> on page 3-1133
0xD08	por_hnf_sam_memregion0	RW	<i>por_hnf_sam_memregion0</i> on page 3-1135
0xD38	por_hnf_sam_memregion0_end_addr	RW	<i>por_hnf_sam_memregion0_end_addr</i> on page 3-1136
0xD10	por_hnf_sam_memregion1	RW	<i>por_hnf_sam_memregion1</i> on page 3-1137
0xD40	por_hnf_sam_memregion1_end_addr	RW	<i>por_hnf_sam_memregion1_end_addr</i> on page 3-1138
0xD18	por_hnf_sam_sn_properties	RW	<i>por_hnf_sam_sn_properties</i> on page 3-1139
0xD20	por_hnf_sam_6sn_nodeid	RW	<i>por_hnf_sam_6sn_nodeid</i> on page 3-1144
0xCE8	por_hnf_sam_sn_properties1	RW	<i>por_hnf_sam_sn_properties1</i> on page 3-1145
0xD30	por_hnf_sam_sn_properties2	RW	<i>por_hnf_sam_sn_properties2</i> on page 3-1148
0x2304+768+896	por_hnf_cml_port_aggr_grp_5-4_add_mask	RW	<i>por_hnf_cml_port_aggr_grp_5-4_add_mask</i> on page 3-1151
0x2304+768+896	por_hnf_cml_port_aggr_grp_5-31_add_mask	RW	<i>por_hnf_cml_port_aggr_grp_5-31_add_mask</i> on page 3-1153
0x2304+768+944	por_hnf_cml_port_aggr_grp_reg_2-12	RW	<i>por_hnf_cml_port_aggr_grp_reg_2-12</i> on page 3-1155
0xFD0	por_hnf_cml_port_aggr_ctrl_reg	RW	<i>por_hnf_cml_port_aggr_ctrl_reg</i> on page 3-1158
0x2304+768+22016	por_hnf_cml_port_aggr_ctrl_reg_1-6	RW	<i>por_hnf_cml_port_aggr_ctrl_reg_1-6</i> on page 3-1161
0xF50	por_hnf_abf_lo_addr	RW	<i>por_hnf_abf_lo_addr</i> on page 3-1164
0xF58	por_hnf_abf_hi_addr	RW	<i>por_hnf_abf_hi_addr</i> on page 3-1165
0xF60	por_hnf_abf_pr	RW	<i>por_hnf_abf_pr</i> on page 3-1166
0xF68	por_hnf_abf_sr	RO	<i>por_hnf_abf_sr</i> on page 3-1167
0x1000	por_hnf_cbusy_write_limit_ctl	RW	<i>por_hnf_cbusy_write_limit_ctl</i> on page 3-1168
0x1008	por_hnf_cbusy_resp_ctl	RW	<i>por_hnf_cbusy_resp_ctl</i> on page 3-1169
0x1010	por_hnf_cbusy_sn_ctl	RW	<i>por_hnf_cbusy_sn_ctl</i> on page 3-1171
0x1020	por_hnf_pocq_alloc_class_dedicated	RW	<i>por_hnf_pocq_alloc_class_dedicated</i> on page 3-1172

Table 3-11 HNF register summary (continued)

Offset	Name	Type	Description
0x1028	por_hnf_pocq_alloc_class_max_allowed	RW	<i>por_hnf_pocq_alloc_class_max_allowed</i> on page 3-1173
0x1030	por_hnf_pocq_alloc_class_contented_min	RW	<i>por_hnf_pocq_alloc_class_contented_min</i> on page 3-1175
0x1038	por_hnf_class_ctl	RW	<i>por_hnf_class_ctl</i> on page 3-1176
0x1040	por_hnf_pocq_qos_class_ctl	RW	<i>por_hnf_pocq_qos_class_ctl</i> on page 3-1177
0x1050	por_hnf_class_pocq_arb_weight_ctl	RW	<i>por_hnf_class_pocq_arb_weight_ctl</i> on page 3-1178
0x1058	por_hnf_class_retry_weight_ctl	RW	<i>por_hnf_class_retry_weight_ctl</i> on page 3-1180
0x16'h1060+#[[0,1,2,...,62,63]]*8	por_hnf_rnf_class_weight_0-63	RW	<i>por_hnf_rnf_class_weight_0-63</i> on page 3-1181
0x16'h1260+#[[0,1,2,...,14,15]]*8	por_hnf_rni_class_weight_0-15	RW	<i>por_hnf_rni_class_weight_0-15</i> on page 3-1182
0x16'h12E0+#[[0,1,2,...,14,15]]*8	por_hnf_rnd_class_weight_0-15	RW	<i>por_hnf_rnd_class_weight_0-15</i> on page 3-1183
0x16'h1360+#[[0,1,2,...,14,15]]*8	por_hnf_cxha_class_weight_0-15	RW	<i>por_hnf_cxha_class_weight_0-15</i> on page 3-1184
0xFE0	por_hnf_partner_scratch_reg0	RW	<i>por_hnf_partner_scratch_reg0</i> on page 3-1186
0xFE8	por_hnf_partner_scratch_reg1	RW	<i>por_hnf_partner_scratch_reg1</i> on page 3-1187
0xB80	por_hnf_cfg_slcsf_dbgrd	WO	<i>por_hnf_cfg_slcsf_dbgrd</i> on page 3-1187
0xB88	por_hnf_slc_cache_access_slc_tag	RO	<i>por_hnf_slc_cache_access_slc_tag</i> on page 3-1189
0xB90	por_hnf_slc_cache_access_slc_tag1	RO	<i>por_hnf_slc_cache_access_slc_tag1</i> on page 3-1190
0xB98	por_hnf_slc_cache_access_slc_data	RO	<i>por_hnf_slc_cache_access_slc_data</i> on page 3-1190
0xBC0	por_hnf_slc_cache_access_slc_mte_tag	RO	<i>por_hnf_slc_cache_access_slc_mte_tag</i> on page 3-1191
0xBA0	por_hnf_slc_cache_access_sf_tag	RO	<i>por_hnf_slc_cache_access_sf_tag</i> on page 3-1192
0xBA8	por_hnf_slc_cache_access_sf_tag1	RO	<i>por_hnf_slc_cache_access_sf_tag1</i> on page 3-1193
0xBB0	por_hnf_slc_cache_access_sf_tag2	RO	<i>por_hnf_slc_cache_access_sf_tag2</i> on page 3-1194
0x2000	por_hnf_pmu_event_sel	RW	<i>por_hnf_pmu_event_sel</i> on page 3-1195
0x2008	por_hnf_pmu_mpam_sel	RW	<i>por_hnf_pmu_mpam_sel</i> on page 3-1200
0x16'h2010+#{8*[0,1,2,3,4,5,6,7]}	por_hnf_pmu_mpam_pardid_mask_0-7	RW	<i>por_hnf_pmu_mpam_pardid_mask_0-7</i> on page 3-1201

Table 3-11 HNF register summary (continued)

Offset	Name	Type	Description
0x16'h3C00+#[[0,1,2,...,62,63]]*32	por_hnf_rn_cluster_0-63_physid_reg0	RW	por_hnf_rn_cluster_0-63_physid_reg0 on page 3-1202
0x16'h3C00+#[[0,1,2,...,126,127]]*32	por_hnf_rn_cluster_64-127_physid_reg0	RW	por_hnf_rn_cluster_64-127_physid_reg0 on page 3-1204
0x16'h3C08+#[[0,1,2,...,126,127]]*32	por_hnf_rn_cluster_0-127_physid_reg1	RW	por_hnf_rn_cluster_0-127_physid_reg1 on page 3-1206
0x16'h3C10+#[[0,1,2,...,126,127]]*32	por_hnf_rn_cluster_0-127_physid_reg2	RW	por_hnf_rn_cluster_0-127_physid_reg2 on page 3-1209
0x16'h3C18+#[[0,1,2,...,126,127]]*32	por_hnf_rn_cluster_0-127_physid_reg3	RW	por_hnf_rn_cluster_0-127_physid_reg3 on page 3-1211
0x16'h5000+#[[0,1,2,...,62,63]]*8	por_hnf_sam_nonhash_cfg1_memregion_2-63	RW	por_hnf_sam_nonhash_cfg1_memregion_2-63 on page 3-1213
0x16'h5200+#[[0,1,2,...,62,63]]*8	por_hnf_sam_nonhash_cfg2_memregion_2-63	RW	por_hnf_sam_nonhash_cfg2_memregion_2-63 on page 3-1215
0x16'h5400+#[[0,1,2,3,4,5,6,7]]*8	por_hnf_sam_htg_cfg1_memregion_0-7	RW	por_hnf_sam_htg_cfg1_memregion_0-7 on page 3-1217
0x16'h5480+#[[0,1,2,3,4,5,6,7]]*8	por_hnf_sam_htg_cfg2_memregion_0-7	RW	por_hnf_sam_htg_cfg2_memregion_0-7 on page 3-1218
0x16'h5500+#[[0,1,2,3,4,5,6,7]]*8	por_hnf_sam_htg_cfg3_memregion_0-7	RW	por_hnf_sam_htg_cfg3_memregion_0-7 on page 3-1219
0x16'h5600+#[[0,1,2,...,14,15]]*8	por_hnf_sam_htg_sn_nodeid_reg_0-15	RW	por_hnf_sam_htg_sn_nodeid_reg_0-15 on page 3-1222
0x16'h5680+#[[0,1,2,...,14,15]]*8	por_hnf_sam_htg_sn_attr_0-15	RW	por_hnf_sam_htg_sn_attr_0-15 on page 3-1223
0x16'h5700+#[[0,1,2,3]]*8	por_hnf_sam_ccg_sa_nodeid_reg_0-3	RW	por_hnf_sam_ccg_sa_nodeid_reg_0-3 on page 3-1227
0x16'h5740+#[[0,1,2,3]]*8	por_hnf_sam_ccg_sa_attr_0-3	RW	por_hnf_sam_ccg_sa_attr_0-3 on page 3-1228
0x16'h5780+#[[0,1]]*8	hnf_generic_regs_0-1	RW	hnf_generic_regs_0-1 on page 3-1232
0x5900	por_hnf_pa2setaddr_slc	RW	por_hnf_pa2setaddr_slc on page 3-1233
0x5908	por_hnf_pa2setaddr_sf	RW	por_hnf_pa2setaddr_sf on page 3-1234
0x5910	por_hnf_pa2setaddr_flex_slc	RW	por_hnf_pa2setaddr_flex_slc on page 3-1235
0x5918	por_hnf_pa2setaddr_flex_sf	RW	por_hnf_pa2setaddr_flex_sf on page 3-1236

3.2.11 MTU register summary

This section lists the MTU registers used in CMN-700.

MTU register summary

The following table shows the *MTU* registers in offset order from the base memory address

Table 3-12 MTU register summary

Offset	Name	Type	Description
0x0	por_mtu_node_info	RO	por_mtu_node_info on page 3-399
0x80	por_mtu_child_info	RO	por_mtu_child_info on page 3-400
0x980	por_mtu_secure_register_groups_override	RW	por_mtu_secure_register_groups_override on page 3-401
0x900	por_mtu_unit_info	RO	por_mtu_unit_info on page 3-402
0xA00	por_mtu_cfg_ctl	RW	por_mtu_cfg_ctl on page 3-403
0xA08	por_mtu_aux_ctl	RW	por_mtu_aux_ctl on page 3-405
0xA30	por_mtu_tc_flush_pr	RW	por_mtu_tc_flush_pr on page 3-406
0xA38	por_mtu_tc_flush_sr	RO	por_mtu_tc_flush_sr on page 3-407
0xA40	por_mtu_tag_addr_ctl	RW	por_mtu_tag_addr_ctl on page 3-408
0xA48	por_mtu_tag_addr_base	RW	por_mtu_tag_addr_base on page 3-409
0x16'hA50+#{8*[0,1,2]}	por_mtu_tag_addr_shutter_0-2	RW	por_mtu_tag_addr_shutter_0-2 on page 3-410
0x3000	por_mtu_errfr	RO	por_mtu_errfr on page 3-417
0x3008	por_mtu_errctlr	RW	por_mtu_errctlr on page 3-419
0x3010	por_mtu_errstatus	W1C	por_mtu_errstatus on page 3-420
0x3018	por_mtu_erraddr	RW	por_mtu_erraddr on page 3-422
0x3020	por_mtu_errmisc	RW	por_mtu_errmisc on page 3-423
0x3100	por_mtu_errfr_NS	RO	por_mtu_errfr_NS on page 3-425
0x3108	por_mtu_errctlr_NS	RW	por_mtu_errctlr_NS on page 3-426
0x3110	por_mtu_errstatus_NS	W1C	por_mtu_errstatus_NS on page 3-428
0x3118	por_mtu_erraddr_NS	RW	por_mtu_erraddr_NS on page 3-429
0x3120	por_mtu_errmisc_NS	RW	por_mtu_errmisc_NS on page 3-431
0x3030	por_mtu_err_inj	RW	por_mtu_err_inj on page 3-432
0xB80	por_mtu_cfg_tc_dbgrd	WO	por_mtu_cfg_tc_dbgrd on page 3-433
0xB88	por_mtu_tc_cache_access_tc_ctl	RO	por_mtu_tc_cache_access_tc_ctl on page 3-435
0xB98	por_mtu_tc_cache_access_tc_data	RO	por_mtu_tc_cache_access_tc_data on page 3-436
0x2000	por_mtu_pmu_event_sel	RW	por_mtu_pmu_event_sel on page 3-436

3.2.12 MXP register summary

This section lists the MXP registers used in CMN-700.

MXP register summary

The following table shows the *MXP* registers in offset order from the base memory address

Table 3-13 MXP register summary

Offset	Name	Type	Description
0x0	por_mxp_node_info	RO	<i>por_mxp_node_info</i> on page 3-966
0x16'h8+#{8*[0,1,2,3,4,5]}	por_mxp_device_port_connect_info_p_0-5	RO	<i>por_mxp_device_port_connect_info_p_0-5</i> on page 3-967
0x38	por_mxp_mesh_port_connect_info_east	RO	<i>por_mxp_mesh_port_connect_info_east</i> on page 3-970
0x40	por_mxp_mesh_port_connect_info_north	RO	<i>por_mxp_mesh_port_connect_info_north</i> on page 3-970
0x16'h48+#{8*[0,1,2,3,4,5]}	por_mxp_device_port_connect_ldid_info_p_0-5	RO	<i>por_mxp_device_port_connect_ldid_info_p_0-5</i> on page 3-971
0x80	por_mxp_child_info	RO	<i>por_mxp_child_info</i> on page 3-972
0x16'h100+#{8*[0,1,2,...,30,31]}	por_mxp_child_pointer_0-31	RO	<i>por_mxp_child_pointer_0-31</i> on page 3-973
0x16'h900+#{16*[0,1,2,3,4,5]}	por_mxp_p_0-5_info	RO	<i>por_mxp_p_0-5_info</i> on page 3-974
0x16'h908+#{16*[0,1,2,3,4,5]}	por_mxp_p_0-5_info_1	RO	<i>por_mxp_p_0-5_info_1</i> on page 3-976
0x960	por_dtm_unit_info	RO	<i>por_dtm_unit_info</i> on page 3-978
0x16'h968+#{8*([1,2,3]-1)}	por_dtm_unit_info_dt_1-3	RO	<i>por_dtm_unit_info_dt_1-3</i> on page 3-979
0x980	por_mxp_secure_register_groups_override	RW	<i>por_mxp_secure_register_groups_override</i> on page 3-980
0xA00	por_mxp_aux_ctl	RW	<i>por_mxp_aux_ctl</i> on page 3-981
0xA08	por_mxp_device_port_ctl	RW	<i>por_mxp_device_port_ctl</i> on page 3-982
0x16'hA10+#{8*[0,1,2,3,4,5]}	por_mxp_p_0-5_mpam_override	RW	<i>por_mxp_p_0-5_mpam_override</i> on page 3-984
0x16'hA40+#{8*[0,1,2,3,4,5]}	por_mxp_p_0-5_ldid_override	RW	<i>por_mxp_p_0-5_ldid_override</i> on page 3-985
0x16'hA80+#{32*[0,1,2,3,4,5]}	por_mxp_p_0-5_qos_control	RW	<i>por_mxp_p_0-5_qos_control</i> on page 3-987
0x16'hA88+#{32*[0,1,2,3,4,5]}	por_mxp_p_0-5_qos_lat_tgt	RW	<i>por_mxp_p_0-5_qos_lat_tgt</i> on page 3-988
0x16'hA90+#{32*[0,1,2,3,4,5]}	por_mxp_p_0-5_qos_lat_scale	RW	<i>por_mxp_p_0-5_qos_lat_scale</i> on page 3-989

Table 3-13 MXP register summary (continued)

Offset	Name	Type	Description
0x16'hA98+#{32*[0,1,2,3,4,5]}	por_mxp_p_0-5_qos_lat_range	RW	<i>por_mxp_p_0-5_qos_lat_range</i> on page 3-991
0x2000	por_mxp_pmu_event_sel	RW	<i>por_mxp_pmu_event_sel</i> on page 3-992
0x3000	por_mxp_errfr	RO	<i>por_mxp_errfr</i> on page 3-994
0x3008	por_mxp_errctlr	RW	<i>por_mxp_errctlr</i> on page 3-996
0x3010	por_mxp_errstatus	W1C	<i>por_mxp_errstatus</i> on page 3-997
0x3028	por_mxp_errmisc	RW	<i>por_mxp_errmisc</i> on page 3-999
0x16'h3030+#{8*[0,1,2,3,4,5]}	por_mxp_p_0-5_byte_par_err_inj	WO	<i>por_mxp_p_0-5_byte_par_err_inj</i> on page 3-1003
0x3100	por_mxp_errfr_NS	RO	<i>por_mxp_errfr_NS</i> on page 3-1003
0x3108	por_mxp_errctlr_NS	RW	<i>por_mxp_errctlr_NS</i> on page 3-1005
0x3110	por_mxp_errstatus_NS	W1C	<i>por_mxp_errstatus_NS</i> on page 3-1006
0x3128	por_mxp_errmisc_NS	RW	<i>por_mxp_errmisc_NS</i> on page 3-1008
0x16'h1C00+#{16*[0,1,2,3,4,5]}	por_mxp_p_0-5_syscoreq_ctl	RW	<i>por_mxp_p_0-5_syscoreq_ctl</i> on page 3-1012
0x16'h1C08+#{16*[0,1,2,3,4,5]}	por_mxp_p_0-5_syscoack_status	RO	<i>por_mxp_p_0-5_syscoack_status</i> on page 3-1013
0x2100	por_dtm_control	RW	<i>por_dtm_control</i> on page 3-1014
0x2118	por_dtm_fifo_entry_ready	W1C	<i>por_dtm_fifo_entry_ready</i> on page 3-1015
0x16'h2120+#{24*[0,1,2,3]}	por_dtm_fifo_entry_0-3_0	RO	<i>por_dtm_fifo_entry_0-3_0</i> on page 3-1016
0x16'h2128+#{24*[0,1,2,3]}	por_dtm_fifo_entry_0-3_1	RO	<i>por_dtm_fifo_entry_0-3_1</i> on page 3-1017
0x16'h2130+#{24*[0,1,2,3]}	por_dtm_fifo_entry_0-3_2	RO	<i>por_dtm_fifo_entry_0-3_2</i> on page 3-1018
0x16'h21A0+#{24*[0,1,2,3]}	por_dtm_wp_0-3_config	RW	<i>por_dtm_wp_0-3_config</i> on page 3-1019
0x16'h21A8+#{24*[0,1,2,3]}	por_dtm_wp_0-3_val	RW	<i>por_dtm_wp_0-3_val</i> on page 3-1021
0x16'h21B0+#{24*[0,1,2,3]}	por_dtm_wp_0-3_mask	RW	<i>por_dtm_wp_0-3_mask</i> on page 3-1022
0x2200	por_dtm_pmsicr	RW	<i>por_dtm_pmsicr</i> on page 3-1023
0x2208	por_dtm_pmsirr	RW	<i>por_dtm_pmsirr</i> on page 3-1024
0x2210	por_dtm_pmu_config	RW	<i>por_dtm_pmu_config</i> on page 3-1025
0x2220	por_dtm_pmevcnt	RW	<i>por_dtm_pmevcnt</i> on page 3-1029

Table 3-13 MXP register summary (continued)

Offset	Name	Type	Description
0x2240	por_dtm_pmevcntsr	RW	<i>por_dtm_pmevcntsr</i> on page 3-1030
0x16'h2100+#{5 12*[1,2,3]}	por_dtm_control_dt_1-3	RW	<i>por_dtm_control_dt_1-3</i> on page 3-1031
0x16'h2118+#{5 12*[1,2,3]}	por_dtm_fifo_entry_ready_dt_1-3	W1C	<i>por_dtm_fifo_entry_ready_dt_1-3</i> on page 3-1032
0x16'h2120+#{2 4*([0,1,2,...,1 0,11)%4)}+ #{512*(([0,1,2,...,10,11]/4)+1)}	por_dtm_fifo_entry_0-11	RO	<i>por_dtm_fifo_entry_0-11</i> on page 3-1033
0x16'h2128+#{2 4*([0,1,2,...,1 0,11)%4)}+ #{512*(([0,1,2,...,10,11]/4)+1)}	por_dtm_fifo_entry_0-11	RO	<i>por_dtm_fifo_entry_0-11</i> on page 3-1033
0x16'h2130+#{2 4*([0,1,2,...,1 0,11)%4)}+ #{512*(([0,1,2,...,10,11]/4)+1)}	por_dtm_fifo_entry_0-11	RO	<i>por_dtm_fifo_entry_0-11</i> on page 3-1033
0x16'h21A0+#{2 4*([0,1,2,...,1 0,11)%4)}+ #{512*(([0,1,2,...,10,11]/4)+1)}	por_dtm_wp_0-11	RW	<i>por_dtm_wp_0-11</i> on page 3-1036
0x16'h21A8+#{2 4*([0,1,2,...,1 0,11)%4)}+ #{512*(([0,1,2,...,10,11]/4)+1)}	por_dtm_wp_0-11	RW	<i>por_dtm_wp_0-11</i> on page 3-1036
0x16'h21B0+#{2 4*([0,1,2,...,1 0,11)%4)}+ #{512*(([0,1,2,...,10,11]/4)+1)}	por_dtm_wp_0-11	RW	<i>por_dtm_wp_0-11</i> on page 3-1036
0x16'h2200+#{5 12*[1,2,3]}	por_dtm_pmsicr_dt_1-3	RW	<i>por_dtm_pmsicr_dt_1-3</i> on page 3-1038
0x16'h2208+#{5 12*[1,2,3]}	por_dtm_pmsirr_dt_1-3	RW	<i>por_dtm_pmsirr_dt_1-3</i> on page 3-1039

Table 3-13 MXP register summary (continued)

Offset	Name	Type	Description
0x16'h2210+#{5 12*[1,2,3]}	por_dtm_pmu_config_dt_1-3	RW	<i>por_dtm_pmu_config_dt_1-3</i> on page 3-1040
0x16'h2220+#{5 12*[1,2,3]}	por_dtm_pmevcnt_dt_1-3	RW	<i>por_dtm_pmevcnt_dt_1-3</i> on page 3-1045
0x16'h2240+#{5 12*[1,2,3]}	por_dtm_pmevcnts_r_dt_1-3	RW	<i>por_dtm_pmevcnts_r_dt_1-3</i> on page 3-1046
0x16'hC00+#{8 *[0,1,2,...,14, 15]}	por_mxp_multi_mesh_chn_sel_0-15	RW	<i>por_mxp_multi_mesh_chn_sel_0-15</i> on page 3-1047
0xC80	por_mxp_multi_mesh_chn_ctrl	RW	<i>por_mxp_multi_mesh_chn_ctrl</i> on page 3-1050
0x16'hc90+#{8 *[0,1,2,3,4,5, 6,7]}	por_mxp_xy_override_sel_0-7	RW	<i>por_mxp_xy_override_sel_0-7</i> on page 3-1051
0x16'hCD0+#{32 *[0,1,2,3,4,5] }	por_mxp_p_0-5_pa2setaddr_slc	RW	<i>por_mxp_p_0-5_pa2setaddr_slc</i> on page 3-1053
0x16'hCD8+#{32 *[0,1,2,3,4,5] }	por_mxp_p_0-5_pa2setaddr_sf	RW	<i>por_mxp_p_0-5_pa2setaddr_sf</i> on page 3-1055
0x16'hCE0+#{32 *[0,1,2,3,4,5] }	por_mxp_p_0-5_pa2setaddr_flex_slc	RW	<i>por_mxp_p_0-5_pa2setaddr_flex_slc</i> on page 3-1056
0x16'hCE8+#{32 *[0,1,2,3,4,5] }	por_mxp_p_0-5_pa2setaddr_flex_sf	RW	<i>por_mxp_p_0-5_pa2setaddr_flex_sf</i> on page 3-1057

3.2.13 HN-F MPAM_NS register summary

This section lists the HN-F MPAM_NS registers used in CMN-700.

HNF_MPAM_NS register summary

The following table shows the *HNF_MPAM_NS* registers in offset order from the base memory address

Table 3-14 HNF_MPAM_NS register summary

Offset	Name	Type	Description
0x0	por_hnf_mpam_ns_node_info	RO	<i>por_hnf_mpam_ns_node_info</i> on page 3-902
0x80	por_hnf_mpam_ns_child_info	RO	<i>por_hnf_mpam_ns_child_info</i> on page 3-903
0x1000	por_hnf_mpam_idr	RO	<i>por_hnf_mpam_idr</i> on page 3-904
0x1018	por_hnf_mpam_iidr	RO	<i>por_hnf_mpam_iidr</i> on page 3-906
0x1020	por_hnf_mpam_aidr	RO	<i>por_hnf_mpam_aidr</i> on page 3-907
0x1028	por_hnf_mpam_impl_idr	RO	<i>por_hnf_mpam_impl_idr</i> on page 3-908
0x1030	por_hnf_mpam_cpor_idr	RO	<i>por_hnf_mpam_cpor_idr</i> on page 3-909
0x1038	por_hnf_mpam_ccap_idr	RO	<i>por_hnf_mpam_ccap_idr</i> on page 3-910

Table 3-14 HNF_MPAM_NS register summary (continued)

Offset	Name	Type	Description
0x1040	por_hnf_mpam_mbw_idr	RO	<i>por_hnf_mpam_mbw_idr</i> on page 3-910
0x1048	por_hnf_mpam_pri_idr	RO	<i>por_hnf_mpam_pri_idr</i> on page 3-912
0x1050	por_hnf_mpam_partid_nrw_idr	RO	<i>por_hnf_mpam_partid_nrw_idr</i> on page 3-914
0x1080	por_hnf_mpam_msmon_idr	RO	<i>por_hnf_mpam_msmon_idr</i> on page 3-915
0x1088	por_hnf_mpam_csumon_idr	RO	<i>por_hnf_mpam_csumon_idr</i> on page 3-916
0x1090	por_hnf_mpam_mbwumon_idr	RO	<i>por_hnf_mpam_mbwumon_idr</i> on page 3-917
0x10F0	por_hnf_ns_mpam_ecr	RW	<i>por_hnf_ns_mpam_ecr</i> on page 3-918
0x10F8	por_hnf_ns_mpam_esr	RW	<i>por_hnf_ns_mpam_esr</i> on page 3-919
0x1100	por_hnf_ns_mpamcfg_part_sel	RW	<i>por_hnf_ns_mpamcfg_part_sel</i> on page 3-920
0x1108	por_hnf_ns_mpamcfg_cmax	RW	<i>por_hnf_ns_mpamcfg_cmax</i> on page 3-921
0x1200	por_hnf_ns_mpamcfg_mbw_min	RW	<i>por_hnf_ns_mpamcfg_mbw_min</i> on page 3-922
0x1208	por_hnf_ns_mpamcfg_mbw_max	RW	<i>por_hnf_ns_mpamcfg_mbw_max</i> on page 3-923
0x1220	por_hnf_ns_mpamcfg_mbw_winwd	RW	<i>por_hnf_ns_mpamcfg_mbw_winwd</i> on page 3-924
0x1400	por_hnf_ns_mpamcfg_pri	RW	<i>por_hnf_ns_mpamcfg_pri</i> on page 3-925
0x1500	por_hnf_ns_mpamcfg_mbw_prop	RW	<i>por_hnf_ns_mpamcfg_mbw_prop</i> on page 3-926
0x1600	por_hnf_ns_mpamcfg_intpartid	RW	<i>por_hnf_ns_mpamcfg_intpartid</i> on page 3-927
0x1800	por_hnf_ns_msmon_cfg_mon_sel	RW	<i>por_hnf_ns_msmon_cfg_mon_sel</i> on page 3-928
0x1808	por_hnf_ns_msmon_capt_evnt	RW	<i>por_hnf_ns_msmon_capt_evnt</i> on page 3-929
0x1810	por_hnf_ns_msmon_cfg_csu_flt	RW	<i>por_hnf_ns_msmon_cfg_csu_flt</i> on page 3-930
0x1818	por_hnf_ns_msmon_cfg_csu_ctl	RW	<i>por_hnf_ns_msmon_cfg_csu_ctl</i> on page 3-931
0x1820	por_hnf_ns_msmon_cfg_mbwu_flt	RW	<i>por_hnf_ns_msmon_cfg_mbwu_flt</i> on page 3-933
0x1828	por_hnf_ns_msmon_cfg_mbwu_ctl	RW	<i>por_hnf_ns_msmon_cfg_mbwu_ctl</i> on page 3-934
0x1840	por_hnf_ns_msmon_csu	RW	<i>por_hnf_ns_msmon_csu</i> on page 3-937
0x1848	por_hnf_ns_msmon_csu_capture	RW	<i>por_hnf_ns_msmon_csu_capture</i> on page 3-938
0x1860	por_hnf_ns_msmon_mbwu	RW	<i>por_hnf_ns_msmon_mbwu</i> on page 3-939
0x1868	por_hnf_ns_msmon_mbwu_capture	RW	<i>por_hnf_ns_msmon_mbwu_capture</i> on page 3-940
0x2000	por_hnf_ns_mpamcfg_cpbm	RW	<i>por_hnf_ns_mpamcfg_cpbm</i> on page 3-941

3.2.14 HN-F MPAM_S register summary

This section lists the HN-F MPAM_S registers used in CMN-700.

HNF_MPAM_S register summary

The following table shows the *HNF_MPAM_S* registers in offset order from the base memory address

Table 3-15 HNF_MPAM_S register summary

Offset	Name	Type	Description
0x0	por_hnf_mpam_s_node_info	RO	<i>por_hnf_mpam_s_node_info</i> on page 3-737
0x80	por_hnf_mpam_s_child_info	RO	<i>por_hnf_mpam_s_child_info</i> on page 3-738
0x980	por_hnf_mpam_s_secure_register_groups_override	RW	<i>por_hnf_mpam_s_secure_register_groups_override</i> on page 3-739
0x1008	por_hnf_mpam_sidr	RO	<i>por_hnf_mpam_sidr</i> on page 3-740
0x10F0	por_hnf_s_mpam_ecr	RW	<i>por_hnf_s_mpam_ecr</i> on page 3-741
0x10F8	por_hnf_s_mpam_esr	RW	<i>por_hnf_s_mpam_esr</i> on page 3-742
0x1100	por_hnf_s_mpamcfg_part_sel	RW	<i>por_hnf_s_mpamcfg_part_sel</i> on page 3-743
0x1108	por_hnf_s_mpamcfg_cmax	RW	<i>por_hnf_s_mpamcfg_cmax</i> on page 3-744
0x1200	por_hnf_s_mpamcfg_mbw_min	RW	<i>por_hnf_s_mpamcfg_mbw_min</i> on page 3-745
0x1208	por_hnf_s_mpamcfg_mbw_max	RW	<i>por_hnf_s_mpamcfg_mbw_max</i> on page 3-746
0x1220	por_hnf_s_mpamcfg_mbw_winwd	RW	<i>por_hnf_s_mpamcfg_mbw_winwd</i> on page 3-747
0x1400	por_hnf_s_mpamcfg_pri	RW	<i>por_hnf_s_mpamcfg_pri</i> on page 3-748
0x1500	por_hnf_s_mpamcfg_mbw_prop	RW	<i>por_hnf_s_mpamcfg_mbw_prop</i> on page 3-749
0x1600	por_hnf_s_mpamcfg_intpartid	RW	<i>por_hnf_s_mpamcfg_intpartid</i> on page 3-750
0x1800	por_hnf_s_msmon_cfg_mon_sel	RW	<i>por_hnf_s_msmon_cfg_mon_sel</i> on page 3-751
0x1808	por_hnf_s_msmon_capt_evnt	RW	<i>por_hnf_s_msmon_capt_evnt</i> on page 3-752
0x1810	por_hnf_s_msmon_cfg_csu_flt	RW	<i>por_hnf_s_msmon_cfg_csu_flt</i> on page 3-754
0x1818	por_hnf_s_msmon_cfg_csu_ctl	RW	<i>por_hnf_s_msmon_cfg_csu_ctl</i> on page 3-755
0x1820	por_hnf_s_msmon_cfg_mbwu_flt	RW	<i>por_hnf_s_msmon_cfg_mbwu_flt</i> on page 3-757
0x1828	por_hnf_s_msmon_cfg_mbwu_ctl	RW	<i>por_hnf_s_msmon_cfg_mbwu_ctl</i> on page 3-758
0x1840	por_hnf_s_msmon_csu	RW	<i>por_hnf_s_msmon_csu</i> on page 3-760
0x1848	por_hnf_s_msmon_csu_capture	RW	<i>por_hnf_s_msmon_csu_capture</i> on page 3-761
0x1860	por_hnf_s_msmon_mbwu	RW	<i>por_hnf_s_msmon_mbwu</i> on page 3-762
0x1868	por_hnf_s_msmon_mbwu_capture	RW	<i>por_hnf_s_msmon_mbwu_capture</i> on page 3-763
0x2000	por_hnf_s_mpamcfg_cpmb	RW	<i>por_hnf_s_mpamcfg_cpmb</i> on page 3-764

3.2.15 HN-I register summary

This section lists the HN-I registers used in CMN-700.

HNI register summary

The following table shows the *HNI* registers in offset order from the base memory address

Table 3-16 HNI register summary

Offset	Name	Type	Description
0x0	por_hni_node_info	RO	por_hni_node_info on page 3-366
0x80	por_hni_child_info	RO	por_hni_child_info on page 3-367
0x980	por_hni_secure_register_groups_override	RW	por_hni_secure_register_groups_override on page 3-368
0x900	por_hni_unit_info	RO	por_hni_unit_info on page 3-369
0xC00	por_hni_sam_addrregion0_cfg	RW	por_hni_sam_addrregion0_cfg on page 3-371
0xC08	por_hni_sam_addrregion1_cfg	RW	por_hni_sam_addrregion1_cfg on page 3-373
0xC10	por_hni_sam_addrregion2_cfg	RW	por_hni_sam_addrregion2_cfg on page 3-374
0xC18	por_hni_sam_addrregion3_cfg	RW	por_hni_sam_addrregion3_cfg on page 3-376
0xA00	por_hni_cfg_ctl	RW	por_hni_cfg_ctl on page 3-378
0xA08	por_hni_aux_ctl	RW	por_hni_aux_ctl on page 3-379
0x3000	por_hni_errfr	RO	por_hni_errfr on page 3-380
0x3008	por_hni_errctlr	RW	por_hni_errctlr on page 3-382
0x3010	por_hni_errstatus	W1C	por_hni_errstatus on page 3-383
0x3018	por_hni_erraddr	RW	por_hni_erraddr on page 3-385
0x3020	por_hni_errmisc	RW	por_hni_errmisc on page 3-386
0x3100	por_hni_errfr_NS	RO	por_hni_errfr_NS on page 3-388
0x3108	por_hni_errctlr_NS	RW	por_hni_errctlr_NS on page 3-389
0x3110	por_hni_errstatus_NS	W1C	por_hni_errstatus_NS on page 3-391
0x3118	por_hni_erraddr_NS	RW	por_hni_erraddr_NS on page 3-392
0x3120	por_hni_errmisc_NS	RW	por_hni_errmisc_NS on page 3-393
0x2000	por_hni_pmu_event_sel	RW	por_hni_pmu_event_sel on page 3-395
0x2008	por_hnp_pmu_event_sel	RW	por_hnp_pmu_event_sel on page 3-397

3.2.16 RN-D register summary

This section lists the RN-D registers used in CMN-700.

RND register summary

The following table shows the RND registers in offset order from the base memory address

Table 3-17 RND register summary

Offset	Name	Type	Description
0x0	por_rnd_node_info	RO	por_rnd_node_info on page 3-249
0x80	por_rnd_child_info	RO	por_rnd_child_info on page 3-250
0x980	por_rnd_secure_register_groups_override	RW	por_rnd_secure_register_groups_override on page 3-251
0x900	por_rnd_unit_info	RO	por_rnd_unit_info on page 3-252

Table 3-17 RND register summary (continued)

Offset	Name	Type	Description
0x908	por_rnd_unit_info2	RO	por_rnd_unit_info2 on page 3-254
0xA00	por_rnd_cfg_ctl	RW	por_rnd_cfg_ctl on page 3-255
0xA08	por_rnd_aux_ctl	RW	por_rnd_aux_ctl on page 3-257
0x16'hA10+#[[0,1,2]*8}	por_rnd_s0-2_port_control	RW	por_rnd_s_0-2_port_control on page 3-260
0x16'hA28+#[[0,1,2]*8}	por_rnd_s0-2_mpam_control	RW	por_rnd_s_0-2_mpam_control on page 3-262
0x16'hA80+#[[0,1,2]*32}	por_rnd_s0-2_qos_control	RW	por_rnd_s_0-2_qos_control on page 3-263
0x16'hA88+#[[0,1,2]*32}	por_rnd_s0-2_qos_lat_tgt	RW	por_rnd_s_0-2_qos_lat_tgt on page 3-265
0x16'hA90+#[[0,1,2]*32}	por_rnd_s0-2_qos_lat_scale	RW	por_rnd_s_0-2_qos_lat_scale on page 3-266
0x16'hA98+#[[0,1,2]*32}	por_rnd_s0-2_qos_lat_range	RW	por_rnd_s_0-2_qos_lat_range on page 3-268
0x2000	por_rnd_pmu_event_sel	RW	por_rnd_pmu_event_sel on page 3-269
0x1C00	por_rnd_syscoreq_ctl	RW	por_rnd_syscoreq_ctl on page 3-273
0x1C08	por_rnd_syscoack_status	RO	por_rnd_syscoack_status on page 3-274

3.2.17 RN-I register summary

This section lists the RN-I registers used in CMN-700.

RNI register summary

The following table shows the *RNI* registers in offset order from the base memory address

Table 3-18 RNI register summary

Offset	Name	Type	Description
0x0	por_rni_node_info	RO	por_rni_node_info on page 3-942
0x80	por_rni_child_info	RO	por_rni_child_info on page 3-943
0x980	por_rni_secure_register_groups_override	RW	por_rni_secure_register_groups_override on page 3-944
0x900	por_rni_unit_info	RO	por_rni_unit_info on page 3-945
0x908	por_rni_unit_info2	RO	por_rni_unit_info2 on page 3-947
0xA00	por_rni_cfg_ctl	RW	por_rni_cfg_ctl on page 3-948
0xA08	por_rni_aux_ctl	RW	por_rni_aux_ctl on page 3-950
0x16'hA10+#[[0,1,2]*8}	por_rni_s0-2_port_control	RW	por_rni_s_0-2_port_control on page 3-953
0x16'hA28+#[[0,1,2]*8}	por_rni_s0-2_mpam_control	RW	por_rni_s_0-2_mpam_control on page 3-955

Table 3-18 RNI register summary (continued)

Offset	Name	Type	Description
0x16'hA80+#[0,1,2]*32}	por_rni_s0-2_qos_control	RW	por_rni_s_0-2_qos_control on page 3-956
0x16'hA88+#[0,1,2]*32}	por_rni_s0-2_qos_lat_tgt	RW	por_rni_s_0-2_qos_lat_tgt on page 3-958
0x16'hA90+#[0,1,2]*32}	por_rni_s0-2_qos_lat_scale	RW	por_rni_s_0-2_qos_lat_scale on page 3-959
0x16'hA98+#[0,1,2]*32}	por_rni_s0-2_qos_lat_range	RW	por_rni_s_0-2_qos_lat_range on page 3-961
0x2000	por_rni_pmu_event_sel	RW	por_rni_pmu_event_sel on page 3-962

3.2.18 RN SAM register summary

This section lists the RN SAM registers used in CMN-700.

RNSAM register summary

The following table shows the *RNSAM* registers in offset order from the base memory address

Table 3-19 RNSAM register summary

Offset	Name	Type	Description
0x0	por_rnsam_node_info	RO	por_rnsam_node_info on page 3-636
0x80	por_rnsam_child_info	RO	por_rnsam_child_info on page 3-637
0x980	por_rnsam_secure_register_groups_override	RW	por_rnsam_secure_register_groups_override on page 3-638
0x900	por_rnsam_unit_info	RO	por_rnsam_unit_info on page 3-639
0x908	por_rnsam_unit_info1	RO	por_rnsam_unit_info1 on page 3-640
0x2304+768	non_hash_mem_region_reg_24-63	RW	non_hash_mem_region_reg_24-63 on page 3-642
0x2304+768+192	non_hash_mem_region_cfg2_reg_24-63	RW	non_hash_mem_region_cfg2_reg_24-63 on page 3-645
0x2304+768+384	non_hash_tgt_nodeid_16-15	RW	non_hash_tgt_nodeid_16-15 on page 3-647
0x11A0	cml_port_aggr_mode_ctrl_reg	RW	cml_port_aggr_mode_ctrl_reg on page 3-649
0x2304+768+1440	cml_port_aggr_mode_ctrl_reg_4-6	RW	cml_port_aggr_mode_ctrl_reg_4-6 on page 3-651
0x16'hE00+#[8*[0,1,2,3]}	sys_cache_grp_region_0-3	RW	sys_cache_grp_region_0-3 on page 3-656
0x2304+768+512	hashed_tgt_grp_cfg1_region_8-31	RW	hashed_tgt_grp_cfg1_region_8-31 on page 3-658
0x2304+768+9472	hashed_tgt_grp_cfg2_region_0-31	RW	hashed_tgt_grp_cfg2_region_0-31 on page 3-661
0x2304+768+576	sys_cache_grp_secondary_reg_8-3	RW	sys_cache_grp_secondary_reg_8-3 on page 3-662
0x2304+768+576	hashed_target_grp_secondary_cfg1_reg_8-31	RW	hashed_target_grp_secondary_cfg1_reg_8-31 on page 3-665

Table 3-19 RNSAM register summary (continued)

Offset	Name	Type	Description
0x2304+768+9984	hashed_target_grp_secondary_cfg2_reg_0-31	RW	hashed_target_grp_secondary_cfg2_reg_0-31 on page 3-668
0x2304+768+10240	hashed_target_grp_hash_cntl_reg_0-31	RW	hashed_target_grp_hash_cntl_reg_0-31 on page 3-669
0xEA0	sys_cache_group_hn_count	RW	sys_cache_group_hn_count on page 3-671
0x2304+768+672	hashed_target_group_hn_count_reg_2-3	RW	hashed_target_group_hn_count_reg_2-3 on page 3-672
0xEC0	sys_cache_grp_nonhash_nodeid	RW	sys_cache_grp_nonhash_nodeid on page 3-674
0x2304+768+704	hashed_target_grp_nonhash_nodeid_reg_6-6	RW	hashed_target_grp_nonhash_nodeid_reg_6-6 on page 3-676
0x2304+768+768	sys_cache_grp_hn_nodeid_reg_0-15	RW	sys_cache_grp_hn_nodeid_reg_0-15 on page 3-679
0x2304+768+768	hashed_target_grp_hnf_nodeid_reg_32-31	RW	hashed_target_grp_hnf_nodeid_reg_32-31 on page 3-680
0x2304+768+10752	hashed_target_grp_hnp_nodeid_reg_0-15	RW	hashed_target_grp_hnp_nodeid_reg_0-15 on page 3-682
0x1120	sys_cache_grp_cal_mode_reg	RW	sys_cache_grp_cal_mode_reg on page 3-684
0x2304+768+1312	hashed_target_grp_cal_mode_reg_4-7	RW	hashed_target_grp_cal_mode_reg_4-7 on page 3-686
0x1180	sys_cache_grp_hn_cpa_en_reg	RW	sys_cache_grp_hn_cpa_en_reg on page 3-691
0x2304+768+1408	hashed_target_grp_hnf_cpa_en_reg_2-1	RW	hashed_target_grp_hnf_cpa_en_reg_2-1 on page 3-692
0x2304+768+11056	hashed_target_grp_hnp_cpa_en_reg_0-0	RW	hashed_target_grp_hnp_cpa_en_reg_0-0 on page 3-693
0x1190	sys_cache_grp_hn_cpa_grp_reg	RW	sys_cache_grp_hn_cpa_grp_reg on page 3-694
0x2304+768+1424	hashed_target_grp_cpa_grp_reg_2-7	RW	hashed_target_grp_cpa_grp_reg_2-7 on page 3-695
0x2304+768+11200	hashed_target_grp_hnf_device_bound_cfg_reg_0-1	RW	hashed_target_grp_hnf_device_bound_cfg_reg_0-1 on page 3-698
0xE80	rnsam_hash_addr_mask_reg	RW	rnsam_hash_addr_mask_reg on page 3-699
0xE88	rnsam_hash_axi_id_mask_reg	RW	rnsam_hash_axi_id_mask_reg on page 3-700
0xE90	rnsam_region_cmp_addr_mask_reg	RW	rnsam_region_cmp_addr_mask_reg on page 3-701
0x2304+768+1472	cml_port_aggr_grp_6-31_add_mask	RW	cml_port_aggr_grp_6-31_add_mask on page 3-702
0x2304+768+1520	cml_port_aggr_grp_reg_3-12	RW	cml_port_aggr_grp_reg_3-12 on page 3-704
0x1208	cml_port_aggr_ctrl_reg	RW	cml_port_aggr_ctrl_reg on page 3-707
0x2304+768+1544	cml_port_aggr_ctrl_reg_0-6	RW	cml_port_aggr_ctrl_reg_0-6 on page 3-709

Table 3-19 RNSAM register summary (continued)

Offset	Name	Type	Description
0xEB0	sys_cache_grp_sn_attr	RW	sys_cache_grp_sn_attr on page 3-712
0xEB8	sys_cache_grp_sn_attr1	RW	sys_cache_grp_sn_attr1 on page 3-715
0x2304+768+1344	sys_cache_grp_sn_sam_cfg_0-3	RW	sys_cache_grp_sn_sam_cfg_0-3 on page 3-717
0x2304+768+1664	sam_qos_mem_region_reg_0-15	RW	sam_qos_mem_region_reg_0-15 on page 3-719
0x2304+768+1856	sam_qos_mem_region_cfg2_reg_0-15	RW	sam_qos_mem_region_cfg2_reg_0-15 on page 3-720
0x2304+768+13312	sam_scg_0-511	RW	sam_scg_0-511 on page 3-721
0x2304+768+17408	sam_scg_0-511	RW	sam_scg_0-511 on page 3-721
0x2304+768+21504	sam_scg_0-63	RW	sam_scg_0-63 on page 3-724
0x2304+768+22016	sam_scg_0-63	RW	sam_scg_0-63 on page 3-724
0x2304+768+22528	sam_scg_0-63	RW	sam_scg_0-63 on page 3-724
0x2304+768+1024	sys_cache_grp_sn_nodeid_reg_0-31	RW	sys_cache_grp_sn_nodeid_reg_0-31 on page 3-729
0x2304+768+2048	sys_cache_grp_region_0-63	RW	sys_cache_grp_region_0-63 on page 3-730
0x2304+768+23040	sys_cache_grp_hashed_regions_sn_nod eid_reg_0-15	RW	sys_cache_grp_hashed_regions_sn_nod eid_reg_0-15 on page 3-731
0x1100	rnsam_status	RW	rnsam_status on page 3-733
0x1108	gic_mem_region_reg	RW	gic_mem_region_reg on page 3-734
0x2304+768+2560	sam_generic_regs_0-7	RW	sam_generic_regs_0-7 on page 3-736

3.2.19 SBSX register summary

This section lists the SBSX registers used in CMN-700.

SBSX register summary

The following table shows the SBSX registers in offset order from the base memory address

Table 3-20 SBSX register summary

Offset	Name	Type	Description
0x0	por_sbsx_node_info	RO	por_sbsx_node_info on page 3-878
0x80	por_sbsx_child_info	RO	por_sbsx_child_info on page 3-879
0x980	por_sbsx_secure_register_groups_over ride	RW	por_sbsx_secure_register_groups_override on page 3-880

Table 3-20 SBSX register summary (continued)

Offset	Name	Type	Description
0x900	por_sbsx_unit_info	RO	<i>por_sbsx_unit_info</i> on page 3-881
0xA00	por_sbsx_cfg_ctl	RW	<i>por_sbsx_cfg_ctl</i> on page 3-883
0xA08	por_sbsx_aux_ctl	RW	<i>por_sbsx_aux_ctl</i> on page 3-884
0xA18	por_sbsx_cbusy_limit_ctl	RW	<i>por_sbsx_cbusy_limit_ctl</i> on page 3-885
0x3000	por_sbsx_errfr	RO	<i>por_sbsx_errfr</i> on page 3-886
0x3008	por_sbsx_errctlr	RW	<i>por_sbsx_errctlr</i> on page 3-887
0x3010	por_sbsx_errstatus	W1C	<i>por_sbsx_errstatus</i> on page 3-888
0x3018	por_sbsx_erraddr	RW	<i>por_sbsx_erraddr</i> on page 3-890
0x3020	por_sbsx_errmisc	RW	<i>por_sbsx_errmisc</i> on page 3-892
0x3100	por_sbsx_errfr_NS	RO	<i>por_sbsx_errfr_NS</i> on page 3-893
0x3108	por_sbsx_errctlr_NS	RW	<i>por_sbsx_errctlr_NS</i> on page 3-894
0x3110	por_sbsx_errstatus_NS	W1C	<i>por_sbsx_errstatus_NS</i> on page 3-895
0x3118	por_sbsx_erraddr_NS	RW	<i>por_sbsx_erraddr_NS</i> on page 3-897
0x3120	por_sbsx_errmisc_NS	RW	<i>por_sbsx_errmisc_NS</i> on page 3-898
0x2000	por_sbsx_pmu_event_sel	RW	<i>por_sbsx_pmu_event_sel</i> on page 3-900

3.3 Register descriptions

This section contains register descriptions.

This section contains the following subsections:

- [3.3.1 RN-D register descriptions on page 3-249](#).
- [3.3.2 CCG_HA register descriptions on page 3-275](#).
- [3.3.3 CXRA register descriptions on page 3-331](#).
- [3.3.4 HN-I register descriptions on page 3-366](#).
- [3.3.5 MTU register descriptions on page 3-399](#).
- [3.3.6 CXLA register descriptions on page 3-438](#).
- [3.3.7 DN register descriptions on page 3-501](#).
- [3.3.8 CCLA register descriptions on page 3-519](#).
- [3.3.9 CXHA register descriptions on page 3-584](#).
- [3.3.10 RN SAM register descriptions on page 3-636](#).
- [3.3.11 HN-F MPAM_S register descriptions on page 3-737](#).
- [3.3.12 Configuration master register descriptions on page 3-765](#).
- [3.3.13 CCG_RA register descriptions on page 3-806](#).
- [3.3.14 Debug and trace register descriptions on page 3-844](#).
- [3.3.15 SBSX register descriptions on page 3-878](#).
- [3.3.16 HN-F MPAM_NS register descriptions on page 3-902](#).
- [3.3.17 RN-I register descriptions on page 3-942](#).
- [3.3.18 XP register descriptions on page 3-966](#).
- [3.3.19 HN-F register descriptions on page 3-1058](#).

3.3.1 RN-D register descriptions

This section lists the RN-D registers.

por_rnd_node_info

Provides component identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h0
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

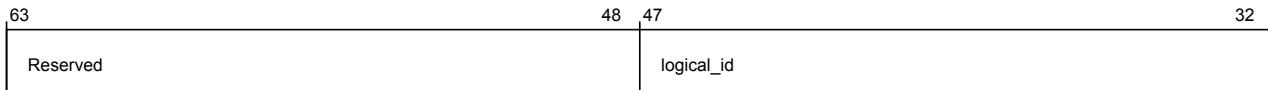


Figure 3-1 por_rnd_por_rnd_node_info (high)

The following table shows the por_rnd_node_info higher register bit assignments.

Table 3-21 por_rnd_por_rnd_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

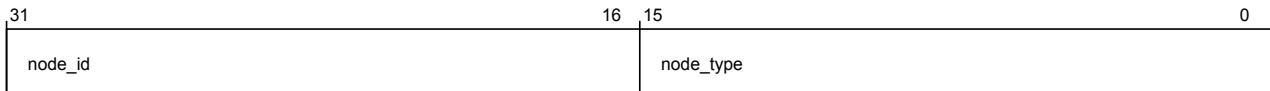


Figure 3-2 por_rnd_por_rnd_node_info (low)

The following table shows the por_rnd_node_info lower register bit assignments.

Table 3-22 por_rnd_por_rnd_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h000D

por_rnd_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

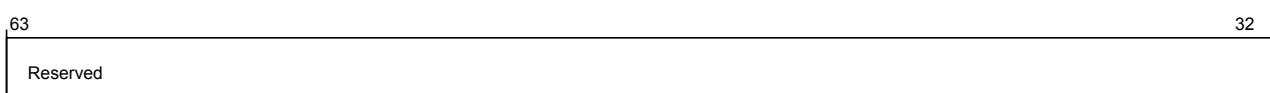


Figure 3-3 por_rnd_por_rnd_child_info (high)

The following table shows the por_rnd_child_info higher register bit assignments.

Table 3-23 por_rnd_por_rnd_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

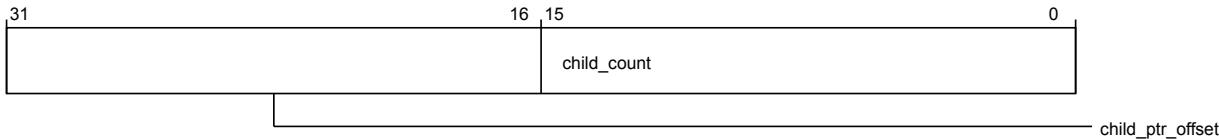


Figure 3-4 por_rnd_por_rnd_child_info (low)

The following table shows the por_rnd_child_info lower register bit assignments.

Table 3-24 por_rnd_por_rnd_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'h0

por_rnd_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

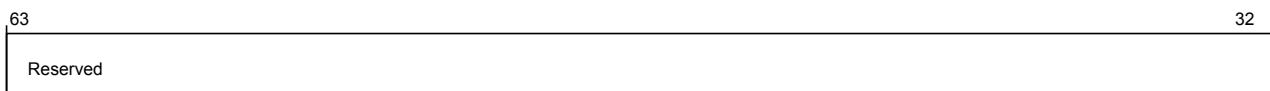


Figure 3-5 por_rnd_por_rnd_secure_register_groups_override (high)

The following table shows the por_rnd_secure_register_groups_override higher register bit assignments.

Table 3-25 por_rnd_por_rnd_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

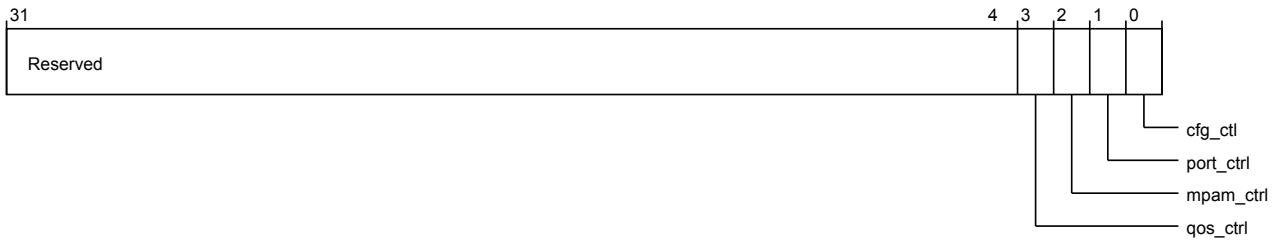


Figure 3-6 por_rnd_por_rnd_secure_register_groups_override (low)

The following table shows the por_rnd_secure_register_groups_override lower register bit assignments.

Table 3-26 por_rnd_por_rnd_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	qos_ctrl	Allows non-secure access to secure QoS control registers	RW	1'b0
2	mpam_ctrl	Allows non-secure access to secure AXI port MPAM override register	RW	1'b0
1	port_ctrl	Allows non-secure access to secure AXI port control registers	RW	1'b0
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_rnd_unit_info

Provides component identification information for RN-D.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h900
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

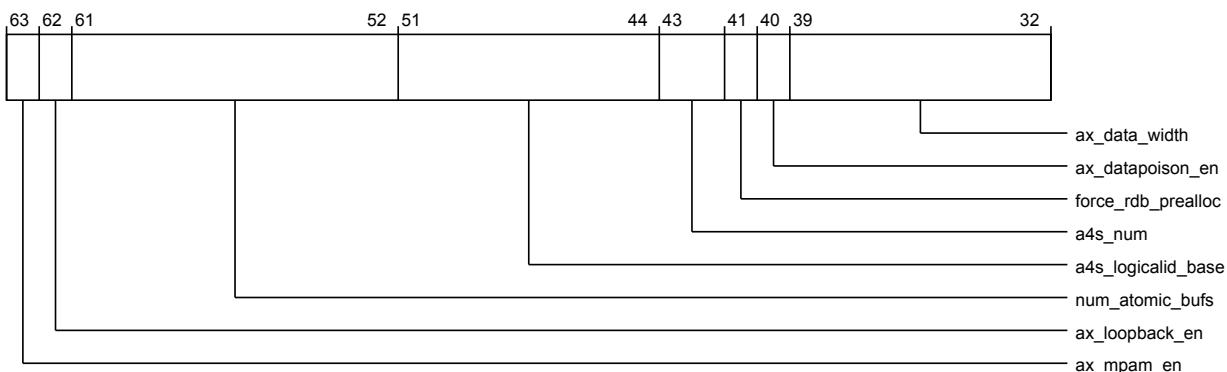


Figure 3-7 por_rnd_por_rnd_unit_info (high)

The following table shows the por_rnd_unit_info higher register bit assignments.

Table 3-27 por_rnd_por_rnd_unit_info (high)

Bits	Field name	Description	Type	Reset
63	ax_mpam_en	MPAM enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
62	ax_loopback_en	LoopBack enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
61:52	num_atomic_bufs	Number of atomic data buffers	RO	Configuration dependent
51:44	a4s_logicalid_base	AXI4Stream interfaces logical ID base	RO	Configuration dependent
43:42	a4s_num	Number of AXI4Stream interfaces present	RO	Configuration dependent
41	force_rdb_prealloc	Force read data buffer preallocation 1'b1: yes 1'b0: no	RO	Configuration dependent
40	ax_datapoison_en	Data Poison enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
39:32	ax_data_width	AXI interface data width in bits	RO	Configuration dependent

The following image shows the lower register bit assignments.

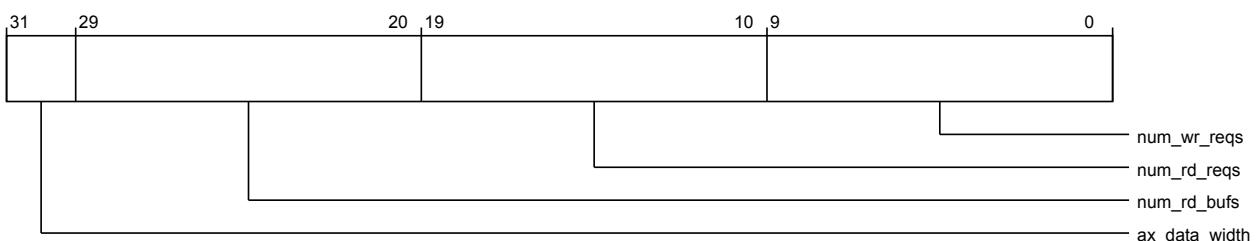


Figure 3-8 por_rnd_por_rnd_unit_info (low)

The following table shows the por_rnd_unit_info lower register bit assignments.

Table 3-28 por_rnd_por_rnd_unit_info (low)

Bits	Field name	Description	Type	Reset
31:30	ax_data_width	AXI interface data width in bits	RO	Configuration dependent
29:20	num_rd_bufs	Number of read data buffers	RO	Configuration dependent
19:10	num_rd_reqs	Number of outstanding read requests	RO	Configuration dependent
9:0	num_wr_reqs	Number of outstanding write requests	RO	Configuration dependent

por_rnd_unit_info2

Provides additional component identification information for RN-D.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h908
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

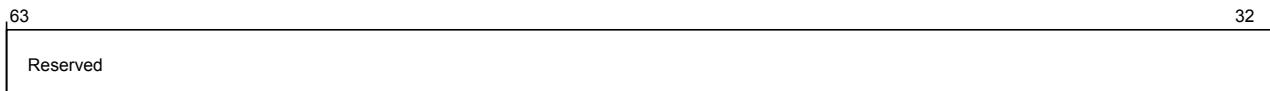


Figure 3-9 por_rnd_por_rnd_unit_info2 (high)

The following table shows the por_rnd_unit_info2 higher register bit assignments.

Table 3-29 por_rnd_por_rnd_unit_info2 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

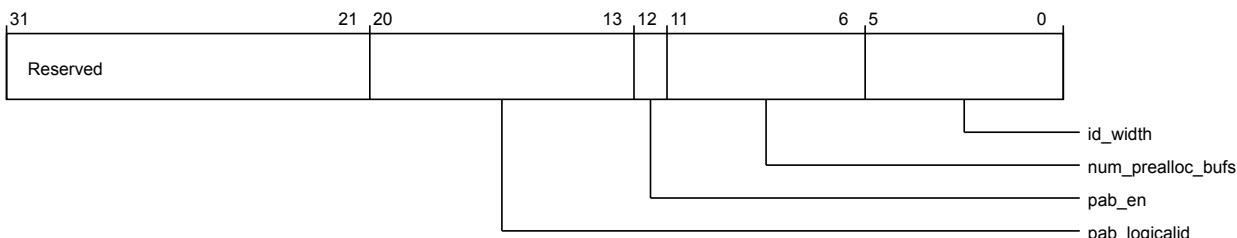


Figure 3-10 por_rnd_por_rnd_unit_info2 (low)

The following table shows the por_rnd_unit_info2 lower register bit assignments.

Table 3-30 por_rnd_por_rnd_unit_info2 (low)

Bits	Field name	Description	Type	Reset
31:21	Reserved	Reserved	RO	-
20:13	pab_logicalid	PUB AUB bridge Logical ID	RO	Configuration dependent
12	pab_en	PUB AUB bridge enable 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent

Table 3-30 por_rnd_por_rnd_unit_info2 (low) (continued)

Bits	Field name	Description	Type	Reset
11:6	num_prealloc_bufs	Number of Pre-allocated Read Data Buffers	RO	Configuration dependent
5:0	id_width	AXI ID width for ACE-Lite slave ports	RO	Configuration dependent

por_rnd_cfg_ctl

Functions as the configuration control register. Specifies the current mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnd_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

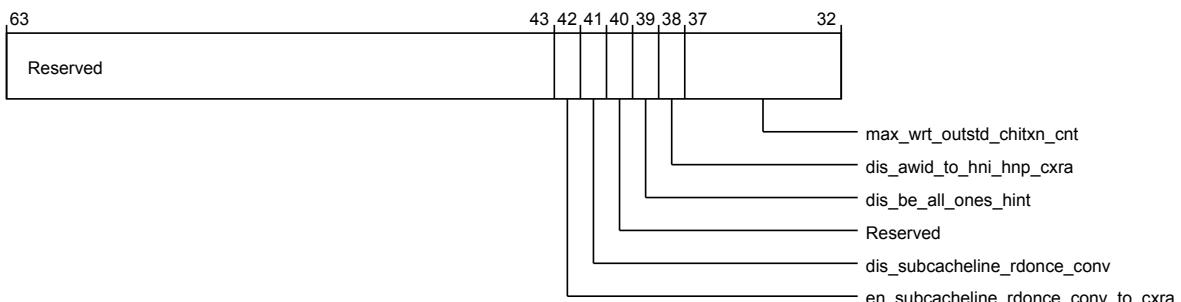


Figure 3-11 por_rnd_por_rnd_cfg_ctl (high)

The following table shows the por_rnd_cfg_ctl higher register bit assignments.

Table 3-31 por_rnd_por_rnd_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:43	Reserved	Reserved	RO	-
42	en_subcacheline_rdonce_conv_to_cxra	If set, enables the conversion of sub-cacheline RdOnce to RdNoSnp for CXRA targets	RW	1'b0
41	dis_subcacheline_rdonce_conv	If set, disables the conversion of sub-cacheline RdOnce to RdNoSnp across all targets	RW	1'b0
40	Reserved	Reserved	RO	-
39	dis_be_all_ones_hint	If set, disables hint to HNF which signals all BE=1's on writes	RW	1'b0

Table 3-31 por_rnd_por_rnd_cfg_ctl (high) (continued)

Bits	Field name	Description	Type	Reset
38	dis_awid_to_hni_hnp_cxra	If set, disables compressed AWID to HNI and CXRA, also disables compressed AWID based ordering. Set this bit if uniq-ID write performance is needed.	RW	1'b0
37:32	max_wrt_outstd_chitxn_cnt	Maximum number of outstanding writes allowed on CHI-side	RW	Configuration dependent

The following image shows the lower register bit assignments.

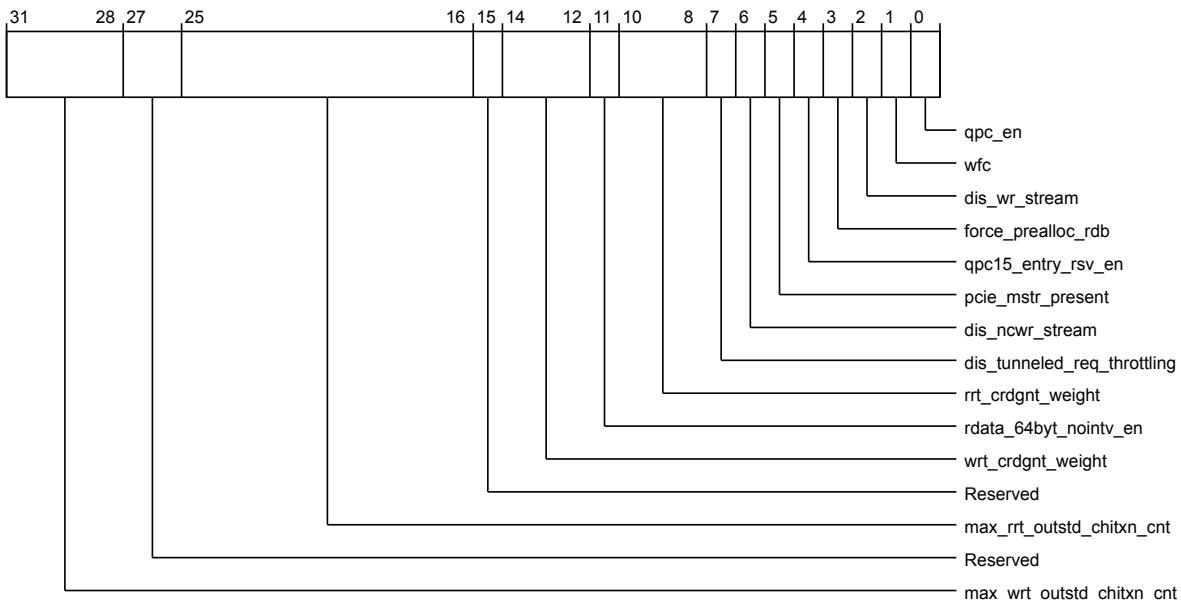


Figure 3-12 por_rnd_por_rnd_cfg_ctl (low)

The following table shows the por_rnd_cfg_ctl lower register bit assignments.

Table 3-32 por_rnd_por_rnd_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	max_wrt_outstd_chitxn_cnt	Maximum number of outstanding writes allowed on CHI-side	RW	Configuration dependent
27:26	Reserved	Reserved	RO	-
25:16	max_rrt_outstd_chitxn_cnt	Maximum number of outstanding reads allowed on CHI-side	RW	Configuration dependent
15	Reserved	Reserved	RO	-
14:12	wrt_crdgnt_weight	Determines weight of credit grant allocated to retried writes in presence of pending retried reads	RW	3'b001
11	rdata_64byt_nointv_en	Enables no interleaving property on normal memory read data within 64B granule when set	RW	1'b1

Table 3-32 por_rnd_por_rnd_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
10:8	rrt_crdgnt_weight	Determines weight of credit grant allocated to retried reads in presence of pending retried writes	RW	3'b100
7	dis_tunneled_req_throttling	Disables retry based throttling of tunneled write requests	RW	1'b0
6	dis_ncwr_stream	Disables streaming of ordered non-cacheable writes when set	RW	1'b0
5	pcie_mstr_present	Indicates PCIe master is present; must be set if PCIe master is present upstream of RN-I or RN-D	RW	1'b0
4	qpc15_entry_rsv_en	Enables QPC15 entry reservation 1'b1: Reserves tracker entry for QoS15 requests 1'b0: Does not reserve tracker entry for QoS15 requests NOTE: Only valid and applicable when por_rnd_qpc_en is set	RW	1'b0
3	force_prealloc_rdb	When set, all reads from the RN-D are sent with a preallocated read data buffer	RW	Configuration dependent
2	dis_wr_stream	Disables streaming of ordered writes when set	RW	1'b0
1	wfc	When set, enables waiting for completion (COMP) before dispatching dependent transaction (TXN)	RW	1'b0
0	qpc_en	When set, enables QPC-based scheduling using two QoS priority classes (QoS15 and non-QoS15)	RW	1'b1

por_rnd_aux_ctl

Functions as the auxiliary control register for RN-D.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-13 por_rnd_por_rnd_aux_ctl (high)

The following table shows the por_rnd_aux_ctl higher register bit assignments.

Table 3-33 por_rnd_por_rnd_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

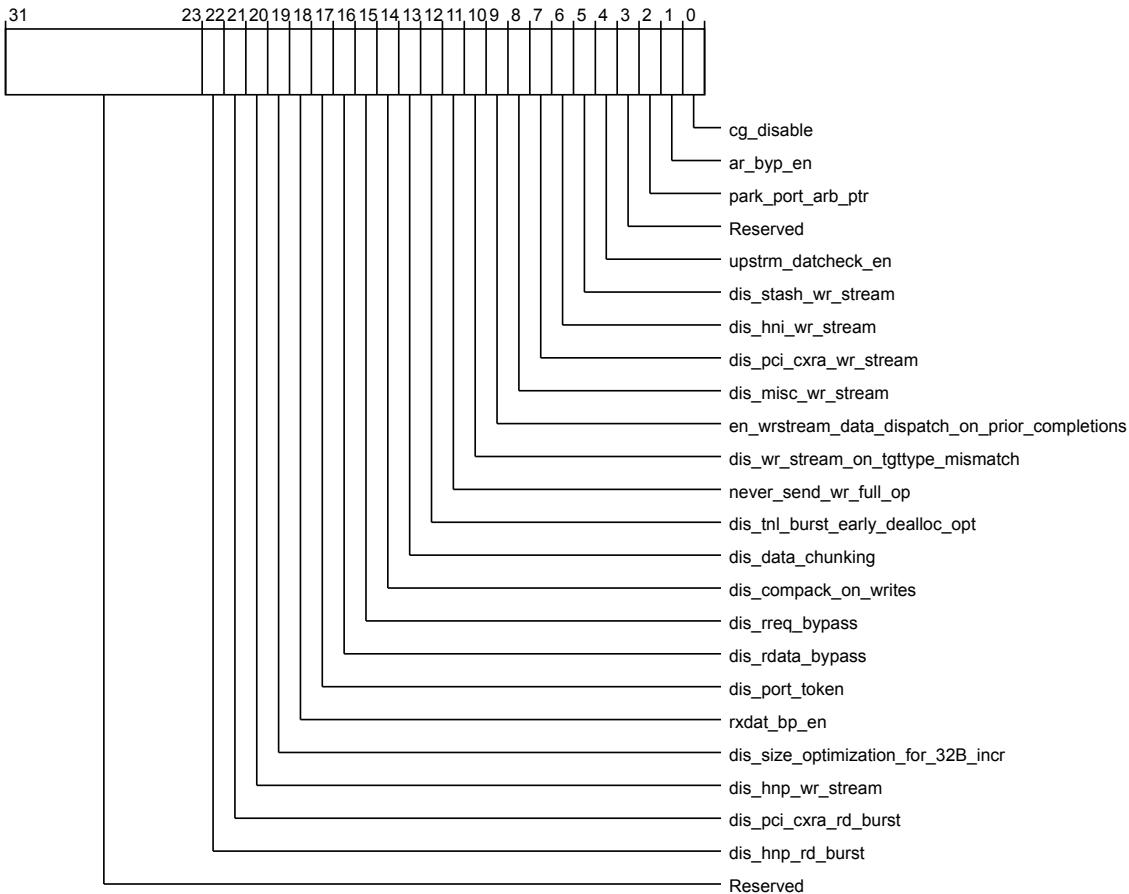


Figure 3-14 por_rnd_por_rnd_aux_ctl (low)

The following table shows the por_rnd_aux_ctl lower register bit assignments.

Table 3-34 por_rnd_por_rnd_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:23	Reserved	Reserved	RO	-
22	dis_hnp_rd_burst	If set, disables read burst to HNP on CHI request flits . Read burst on CHI is supported only in non-decoupled RDB configuration.	RW	1'b0
21	dis_pci_cxra_rd_burst	If set, disables read burst to PCI-CXRA on CHI request flits . Read burst on CHI is supported only in non-decoupled RDB configuration.	RW	1'b0

Table 3-34 por_rnd_por_rnd_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
20	dis_hnp_wr_stream	Disables streaming of ordered writes to HNP when set	RW	1'b0
19	dis_size_optimization_for_32B_incr	If set, disables the size related optimization for a 32B INCR burst (rh-2512)	RW	1'b0
18	rxdat_bp_en	If set, back pressures the rxdat interface when RDB's are not available	RW	1'b0
17	dis_port_token	If set, disables per port reservation in the tracker(rd and wr)	RW	1'b1
16	dis_rdata_bypass	If set, disables read data bypass path; if rxdat_bp_en = 1'b0	RW	1'b0
15	dis_rreq_bypass	If set, disables read request bypass path	RW	1'b0
14	dis_compack_on_writes	If set, disables comp_ack on streaming writes. WrData is used for ordering writes	RW	1'b1
13	dis_data_chunking	If set, disables the data chunking feature	RW	1'b0
12	dis_tnl_burst_early_dealloc_opt	If set, disables the optimization related to early deallocation of tunneled writes for intermediate txns of burst	RW	1'b0
11	never_send_wr_full_op	If set, RNI will never send WR FULL op. All write ops will be of PTL type	RW	1'b0
10	dis_wr_stream_on_tgtype_mismatch	If set, serializes first write when moving from one tgtype to another	RW	1'b0
9	en_wrstream_data_dispatch_on_prior_completions	If set, data dispatch for streaming writes waits for completion of all older writes	RW	1'b0
8	dis_misc_wr_stream	Disables streaming of ordered writes with following attributes when set : Device memory or EWA=0 or Excl=1	RW	1'b0
7	dis_pci_cxra_wr_stream	Disables streaming of ordered writes to PCI-CXRA when set	RW	1'b0
6	dis_hni_wr_stream	Disables streaming of ordered writes to HNI when set	RW	1'b0
5	dis_stash_wr_stream	Disables streaming of ordered WrUniqStash when set	RW	1'b0
4	upstrm_datcheck_en	Upstream supports Datacheck	RW	Configuration dependent
3	Reserved	Reserved	RO	-

Table 3-34 por_rnd_por_rnd_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	park_port_arb_ptr	Parks the AXI port arbitration pointer for Burst	RW	1'b0
1	ar_byp_en	AR bypass enable; enables bypass path in the AR pipeline	RW	1'b1
0	cg_disable	Disables clock gating when set	RW	1'b0

por_rnd_s_0-2_port_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls port S#{index} AXI/ACE slave interface settings.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA10 + # {[0, 1, 2]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnd_secure_register_groups_override.port_ctrl

The following image shows the higher register bit assignments.

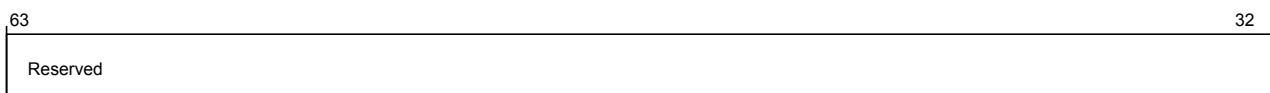


Figure 3-15 por_rnd_por_rnd_s_0-2_port_control (high)

The following table shows the por_rnd_s_0-2_port_control higher register bit assignments.

Table 3-35 por_rnd_por_rnd_s_0-2_port_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

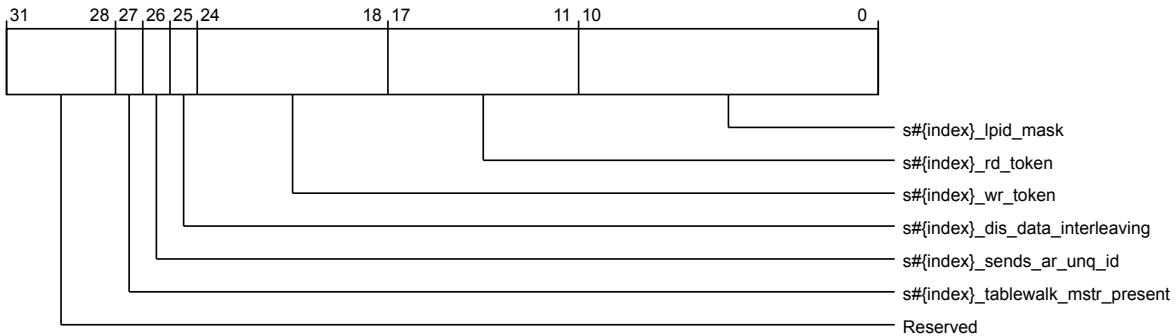


Figure 3-16 por_rnd_por_rnd_s_0-2_port_control (low)

The following table shows the por_rnd_s_0-2_port_control lower register bit assignments.

Table 3-36 por_rnd_por_rnd_s_0-2_port_control (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27	s#{index}_tablewalk_mstr_present	If set, Indicates translation table walk master present such as TCU or GIC	RW	1'b0
26	s#{index}_sends_ar_unq_id	If set, indicates AR transactions on Port#{index} are always Unique ID. This bit for a port must be set to 1 to enable Read Burst on the CHI side of RND. This bit works in conjunction with dis_rd_burst being 0.	RW	1'b0
25	s#{index}_dis_data_interleaving	If set, disables read DATA interleaving on RDATAS#{index} channel. This applies only to RDATA generated as a response to requests on AR channel . This does not apply to RDATA generated as a response to Atomic request on AW channel. I.e. RDATA of an Atomic op, on AW channel, may interleave with RDATA of an AR channel request	RW	1'b0
24:18	s#{index}_wr_token	Port S#{index} reserved token count for AW channel This must be less than the number of Wr requests(RNID_NUM_WR_REQ_PARAM) on AW achnnel	RW	6'b00_0000

Table 3-36 por_rnd_por_rnd_s_0-2_port_control (low) (continued)

Bits	Field name	Description	Type	Reset
17:11	s#{index}_rd_token	Port S#{index} reserved token count for AR channel per slice This should be less than the number of Rd requests(RNID_NUM_RD_REQ_PARAM) per slice on AR achnnel	RW	6'b00_0000
10:0	s#{index}_lpid_mask	Port S#{index} LPID mask LPID[0]: Equal to the result of UnaryOR of BitwiseAND of LPID mask and AXID (LPID[0] = (AXID and mask)); specifies which AXID bit is reflected in the LSB of LPID LPID[2:1]: Equal to port ID[1:0]; the MSB of LPID contains port ID	RW	11'b000_0000_0000

por_rnd_s_0-2_mpam_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls port S#{index} AXI/ACE slave interface MPAM override values

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA28 + # {[0, 1, 2]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnd_secure_register_groups_override.mpam_ctrl

The following image shows the higher register bit assignments.



Figure 3-17 por_rnd_por_rnd_s_0-2_mpam_control (high)

The following table shows the por_rnd_s_0-2_mpam_control higher register bit assignments.

Table 3-37 por_rnd_por_rnd_s_0-2_mpam_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

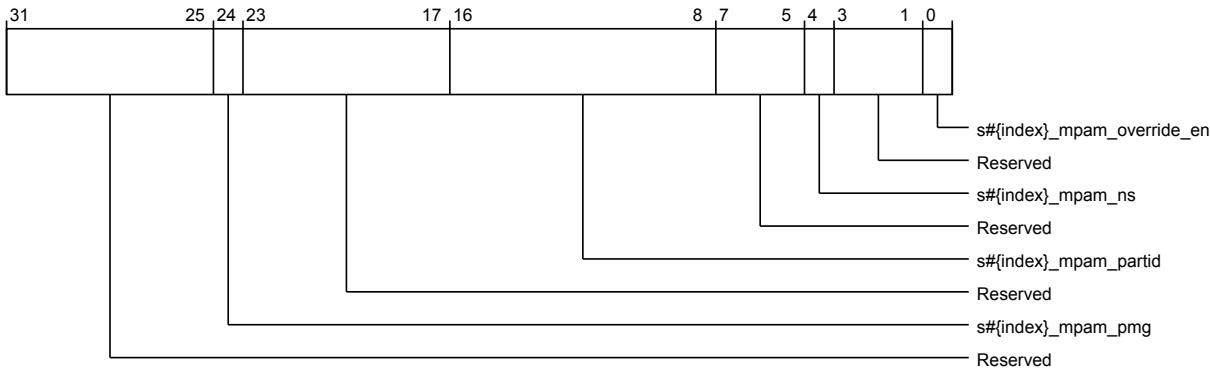


Figure 3-18 por_rnd_por_rnd_s_0-2_mpam_control (low)

The following table shows the por_rnd_s_0-2_mpam_control lower register bit assignments.

Table 3-38 por_rnd_por_rnd_s_0-2_mpam_control (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	s#{index}_mpam_pmg	Port S#{index} MPAM_PMG value	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	s#{index}_mpam_partid	Port S#{index} MPAM_PARTID value	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	s#{index}_mpam_ns	Port S#{index} MPAM_NS value	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	s#{index}_mpam_override_en	Port S#{index} MPAM override en When set, MPAM value on CHI side is driven from MPAM override value in this register. Note that when RNID_AXMPAM_EN_PARAM is set to 0, MPAM override value is always used irrespective of this bit value	RW	1'b0

por_rnd_s_0-2_qos_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls QoS settings for port S#{index} AXI/ACE slave interface.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA80 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnd_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

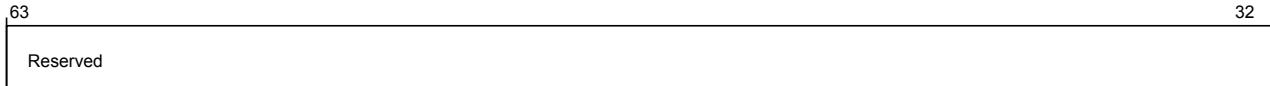


Figure 3-19 por_rnd_por_rnd_s_0-2_qos_control (high)

The following table shows the por_rnd_s_0-2_qos_control higher register bit assignments.

Table 3-39 por_rnd_por_rnd_s_0-2_qos_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

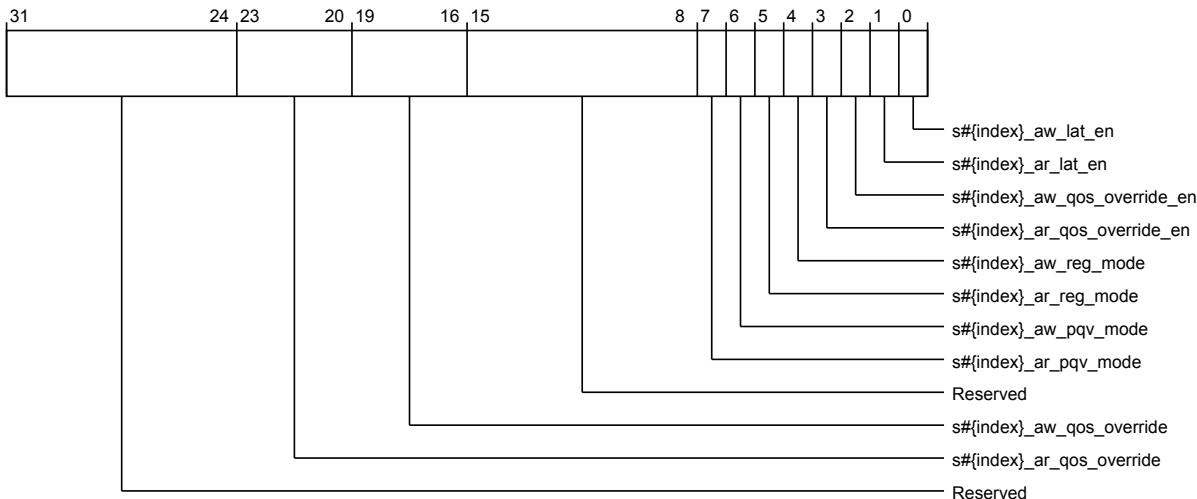


Figure 3-20 por_rnd_por_rnd_s_0-2_qos_control (low)

The following table shows the por_rnd_s_0-2_qos_control lower register bit assignments.

Table 3-40 por_rnd_por_rnd_s_0-2_qos_control (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:20	s#{index}_ar_qos_override	AR QoS override value for port S#{index}	RW	4'b0000
19:16	s#{index}_aw_qos_override	AW QoS override value for port S#{index}	RW	4'b0000
15:8	Reserved	Reserved	RO	-

Table 3-40 por_rnd_por_rnd_s_0-2_qos_control (low) (continued)

Bits	Field name	Description	Type	Reset
7	s#{index}_ar_pqv_mode	Configures the QoS regulator mode for read transactions during period mode 1'b0: Normal mode; QoS value is stable when the master is idle 1'b1: Quiesce high mode; QoS value tends to the maximum value when the master is idle	RW	1'b0
6	s#{index}_aw_pqv_mode	Configures the QoS regulator mode for write transactions during period mode 1'b0: Normal mode; QoS value is stable when the master is idle 1'b1: Quiesce high mode; QoS value tends to the maximum value when the master is idle	RW	1'b0
5	s#{index}_ar_reg_mode	Configures the QoS regulator mode for read transactions 1'b0: Latency mode 1'b1: Period mode; used for bandwidth regulation	RW	1'b0
4	s#{index}_aw_reg_mode	Configures the QoS regulator mode for write transactions 1'b0: Latency mode 1'b1: Period mode; used for bandwidth regulation	RW	1'b0
3	s#{index}_ar_qos_override_en	Enables port S#{index} AR QoS override; when set, allows QoS value on inbound AR transactions to be overridden	RW	1'b0
2	s#{index}_aw_qos_override_en	Enables port S#{index} AW QoS override; when set, allows QoS value on inbound AW transactions to be overridden	RW	1'b0
1	s#{index}_ar_lat_en	Enables port S#{index} AR QoS regulation when set	RW	1'b0
0	s#{index}_aw_lat_en	Enables port S#{index} AW QoS regulation when set	RW	1'b0

por_rnd_s_0-2_qos_lat_tgt

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls QoS target latency (in cycles) for regulations of port S#{index} read and write transactions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA88 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnd_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

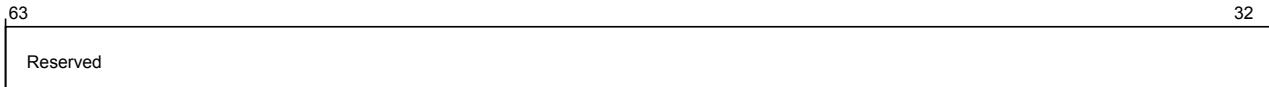


Figure 3-21 por_rnd_por_rnd_s_0-2_qos_lat_tgt (high)

The following table shows the por_rnd_s_0-2_qos_lat_tgt higher register bit assignments.

Table 3-41 por_rnd_por_rnd_s_0-2_qos_lat_tgt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

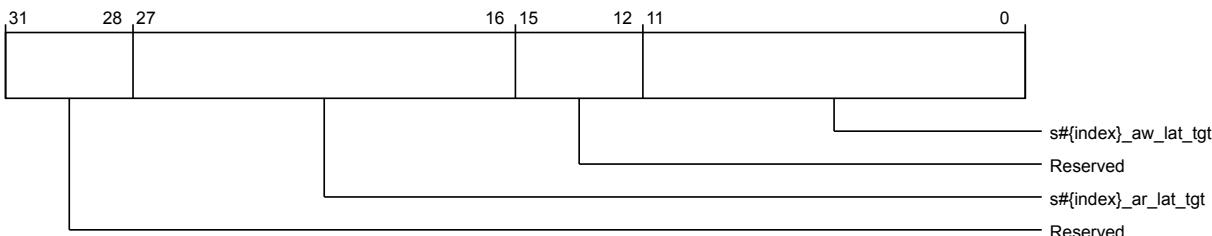


Figure 3-22 por_rnd_por_rnd_s_0-2_qos_lat_tgt (low)

The following table shows the por_rnd_s_0-2_qos_lat_tgt lower register bit assignments.

Table 3-42 por_rnd_por_rnd_s_0-2_qos_lat_tgt (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:16	s#{index}_ar_lat_tgt	Port S#{index} AR channel target latency; a value of 0 corresponds to no regulation	RW	12'h000
15:12	Reserved	Reserved	RO	-
11:0	s#{index}_aw_lat_tgt	Port S#{index} AW channel target latency; a value of 0 corresponds to no regulation	RW	12'h000

por_rnd_s_0-2_qos_lat_scale

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls the QoS target latency scale factor for port S#{index} read and write transactions. This register represents powers of two from the range 2^{-5} to 2^{-12} ; it is used to match a 16-bit integrator.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA90 + # {[0, 1, 2]*32}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_rnd_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

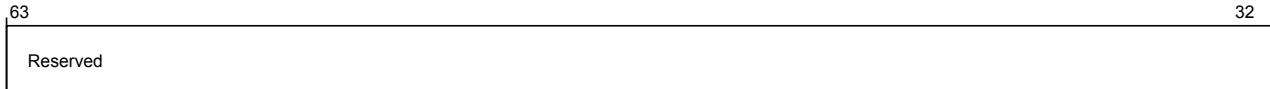


Figure 3-23 por_rnd_por_rnd_s_0-2_qos_lat_scale (high)

The following table shows the por_rnd_s_0-2_qos_lat_scale higher register bit assignments.

Table 3-43 por_rnd_por_rnd_s_0-2_qos_lat_scale (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

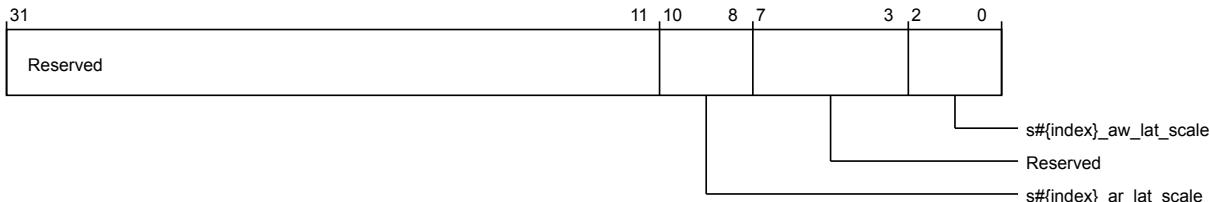


Figure 3-24 por_rnd_por_rnd_s_0-2_qos_lat_scale (low)

The following table shows the por_rnd_s_0-2_qos_lat_scale lower register bit assignments.

Table 3-44 por_rnd_por_rnd_s_0-2_qos_lat_scale (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10:8	s#{index}_ar_lat_scale	Port S#{index} AR QoS scale factor 3'b000: 2^(-5) 3'b001: 2^(-6) 3'b010: 2^(-7) 3'b011: 2^(-8) 3'b100: 2^(-9) 3'b101: 2^(-10) 3'b110: 2^(-11) 3'b111: 2^(-12)	RW	3'h0

Table 3-44 por_rnd_por_rnd_s_0-2_qos_lat_scale (low) (continued)

Bits	Field name	Description	Type	Reset
7:3	Reserved	Reserved	RO	-
2:0	s#{index}_aw_lat_scale	Port S#{index} AW QoS scale factor 3'b000: 2 ⁽⁻⁵⁾ 3'b001: 2 ⁽⁻⁶⁾ 3'b010: 2 ⁽⁻⁷⁾ 3'b011: 2 ⁽⁻⁸⁾ 3'b100: 2 ⁽⁻⁹⁾ 3'b101: 2 ⁽⁻¹⁰⁾ 3'b110: 2 ⁽⁻¹¹⁾ 3'b111: 2 ⁽⁻¹²⁾	RW	3'h0

por_rnd_s_0-2_qos_lat_range

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls the minimum and maximum QoS values generated by the QoS latency regulator for port S#{index} read and write transactions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA98 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnd_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

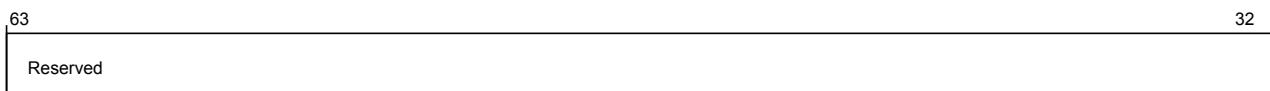


Figure 3-25 por_rnd_por_rnd_s_0-2_qos_lat_range (high)

The following table shows the por_rnd_s_0-2_qos_lat_range higher register bit assignments.

Table 3-45 por_rnd_por_rnd_s_0-2_qos_lat_range (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

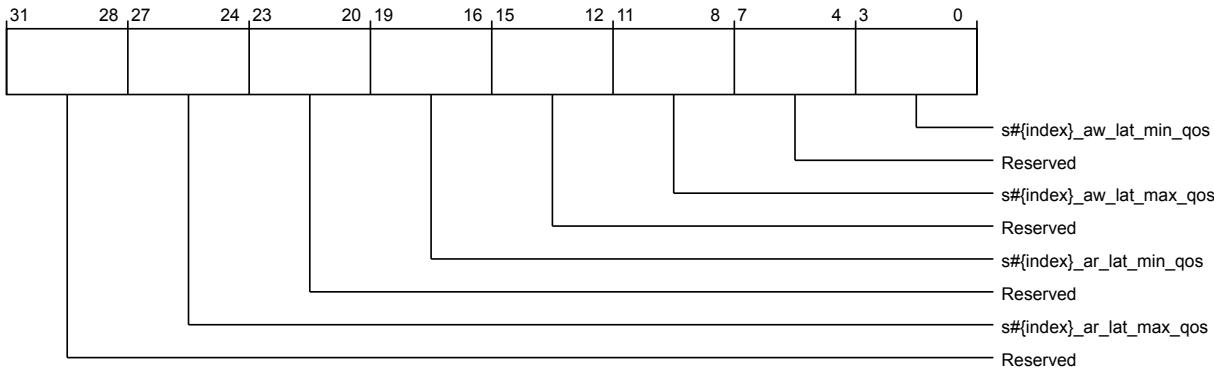


Figure 3-26 por_rnd_por_rnd_s_0-2_qos_lat_range (low)

The following table shows the por_rnd_s_0-2_qos_lat_range lower register bit assignments.

Table 3-46 por_rnd_por_rnd_s_0-2_qos_lat_range (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	s#{index}_ar_lat_max_qos	Port S#{index} AR QoS maximum value	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	s#{index}_ar_lat_min_qos	Port S#{index} AR QoS minimum value	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	s#{index}_aw_lat_max_qos	Port S#{index} AW QoS maximum value	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	s#{index}_aw_lat_min_qos	Port S#{index} AW QoS minimum value	RW	4'h0

por_rnd_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

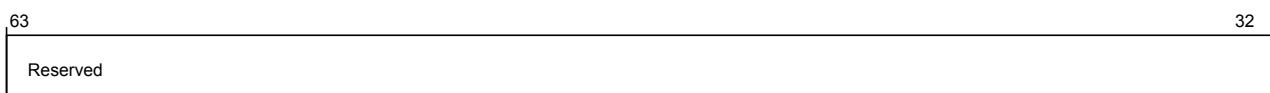


Figure 3-27 por_rnd_por_rnd_pmu_event_sel (high)

The following table shows the por_rnd_pmu_event_sel higher register bit assignments.

Table 3-47 por_rnd_por_rnd_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

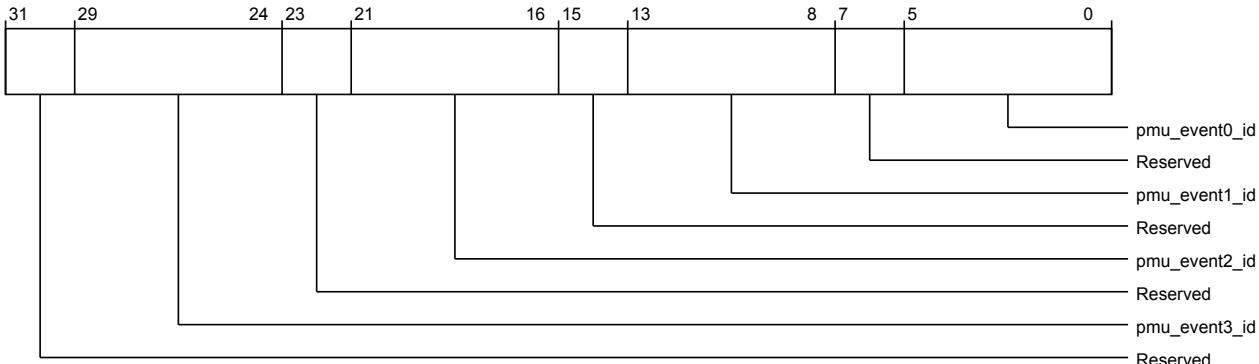


Figure 3-28 por_rnd_por_rnd_pmu_event_sel (low)

The following table shows the por_rnd_pmu_event_sel lower register bit assignments.

Table 3-48 por_rnd_por_rnd_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	RN-D PMU Event 3 ID; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	RN-D PMU Event 2 ID; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	RN-D PMU Event 1 ID; see pmu_event0_id for encodings	RW	6'b0

Table 3-48 por_rnd_por_rnd_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-

Table 3-48 por_rnd_por_rnd_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
5:0	pmu_event0_id	RN-D PMU Event 0 ID 6'h00: No event 6'h01: Port S0 RDataBeats 6'h02: Port S1 RDataBeats 6'h03: Port S2 RDataBeats 6'h04: RXDAT flits received 6'h05: TXDAT flits sent 6'h06: Total TXREQ flits sent 6'h07: Retried TXREQ flits sent 6'h08: RRT occupancy count overflow_slice0 6'h09: WRT occupancy count overflow 6'h0A: Replayed TXREQ flits 6'h0B: WriteCancel sent 6'h0C: Port S0 WDataBeats 6'h0D: Port S1 WDataBeats 6'h0E: Port S2 WDataBeats 6'h0F: RRT allocation 6'h10: WRT allocation 6'h11: PADB occupancy count overflow 6'h12: RPDB occupancy count overflow 6'h13: RRT occupancy count overflow_slice1 6'h14: RRT occupancy count overflow_slice2 6'h15: RRT occupancy count overflow_slice3 6'h16: WRT request throttled 6'h17: RNI backpressure CHI LDB full 6'h18: RRT normal rd req occupancy count overflow_slice0 6'h19: RRT normal rd req occupancy count overflow_slice1 6'h1A: RRT normal rd req occupancy count overflow_slice2 6'h1B: RRT normal rd req occupancy count overflow_slice3 6'h1C: RRT PCIe RD burst req occupancy count overflow_slice0 6'h1D: RRT PCIe RD burst req occupancy count overflow_slice1 6'h1E: RRT PCIe RD burst req occupancy count overflow_slice2 6'h1F: RRT PCIe RD burst req occupancy count overflow_slice3 6'h20: RRT PCIe RD burst allocation 6'h21: Compressed AWID ordering 6'h22: Atomic data buffer allocation 6'h23: Atomic data buffer occupancy	RW	6'b0

por_rnd_syscoreq_ctl

Functions as the RN-D DVM domain control register. Provides a software alternative to hardware SYSCOREQ/SYSCOACK handshake. Works with por_rnd_syscoack_status.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1C00

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

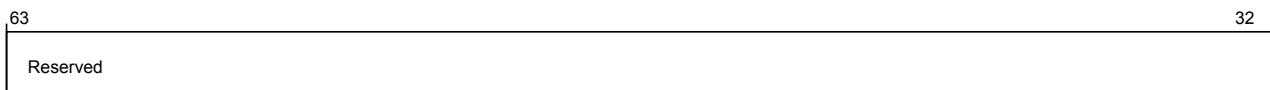


Figure 3-29 por_rnd_por_rnd_syscoreq_ctl (high)

The following table shows the por_rnd_syscoreq_ctl higher register bit assignments.

Table 3-49 por_rnd_por_rnd_syscoreq_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

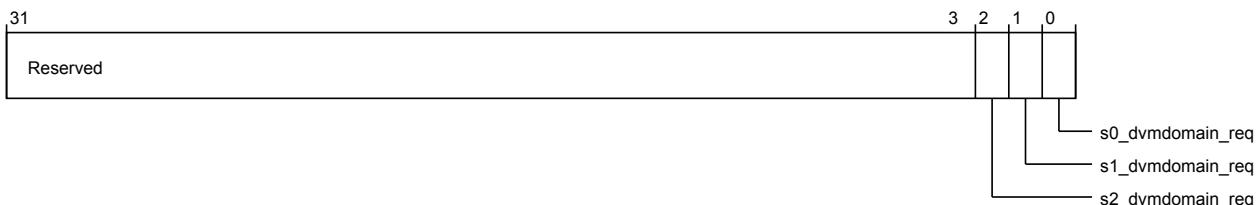


Figure 3-30 por_rnd_por_rnd_syscoreq_ctl (low)

The following table shows the por_rnd_syscoreq_ctl lower register bit assignments.

Table 3-50 por_rnd_por_rnd_syscoreq_ctl (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	s2_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for port S2	RW	1'b0
1	s1_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for port S1	RW	1'b0
0	s0_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for port S0	RW	1'b0

por_rnd_syscoack_status

Functions as the RN-D DVM domain status register. Provides a software alternative to hardware SYSCOREQ/SYSCOACK handshake. Works with por_rnd_syscoreq_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C08

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-31 por_rnd_por_rnd_syscoack_status (high)

The following table shows the por_rnd_syscoack_status higher register bit assignments.

Table 3-51 por_rnd_por_rnd_syscoack_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

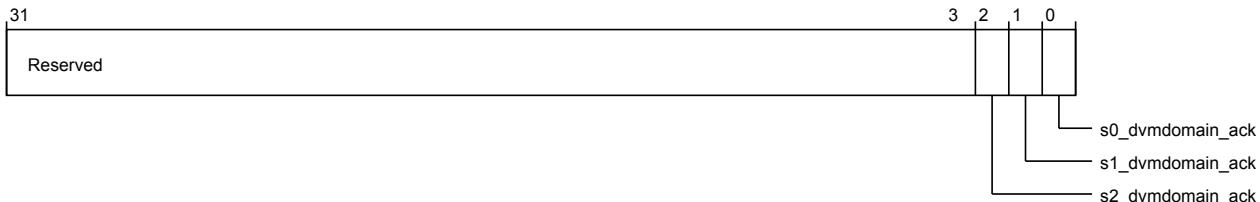


Figure 3-32 por_rnd_por_rnd_syscoack_status (low)

The following table shows the por_rnd_syscoack_status lower register bit assignments.

Table 3-52 por_rnd_por_rnd_syscoack_status (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	s2_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for port S2	RO	1'b0
1	s1_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for port S1	RO	1'b0
0	s0_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for port S0	RO	1'b0

3.3.2 CCG_HA register descriptions

This section lists the CCG_HA registers.

por_ccg_ha_node_info

Provides component identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h0
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

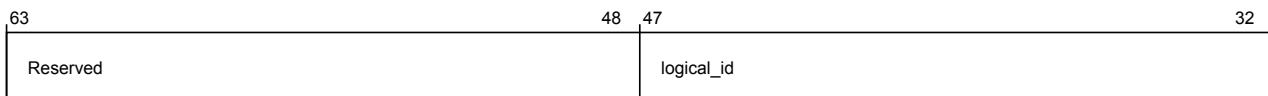


Figure 3-33 por_ccg_ha_por_ccg_ha_node_info (high)

The following table shows the por_ccg_ha_node_info higher register bit assignments.

Table 3-53 por_ccg_ha_por_ccg_ha_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

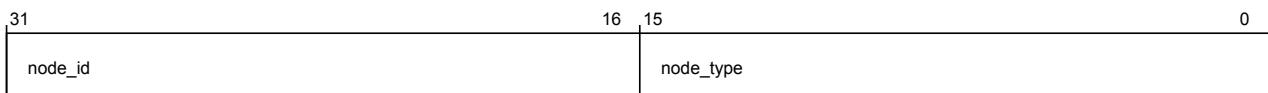


Figure 3-34 por_ccg_ha_por_ccg_ha_node_info (low)

The following table shows the por_ccg_ha_node_info lower register bit assignments.

Table 3-54 por_ccg_ha_por_ccg_ha_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0104

por_ccg_ha_id

Contains the CCIX-assigned HAID.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

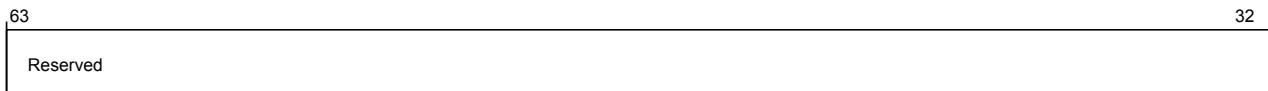


Figure 3-35 por_ccg_ha_por_ccg_ha_id (high)

The following table shows the por_ccg_ha_id higher register bit assignments.

Table 3-55 por_ccg_ha_por_ccg_ha_id (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

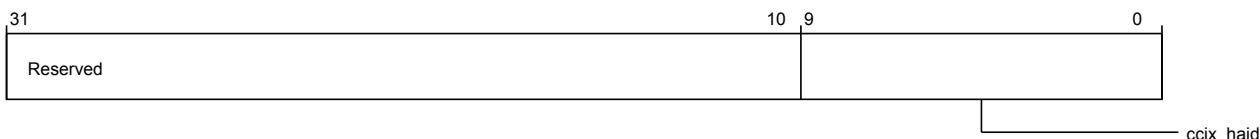


Figure 3-36 por_ccg_ha_por_ccg_ha_id (low)

The following table shows the por_ccg_ha_id lower register bit assignments.

Table 3-56 por_ccg_ha_por_ccg_ha_id (low)

Bits	Field name	Description	Type	Reset
31:10	Reserved	Reserved	RO	-
9:0	ccix_haid	CCIX HAID	RW	10'h0

por_ccg_ha_child_info

Provides component child identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64

Address offset	16'h80
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-37 por_ccg_ha_por_ccg_ha_child_info (high)

The following table shows the por_ccg_ha_child_info higher register bit assignments.

Table 3-57 por_ccg_ha_por_ccg_ha_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

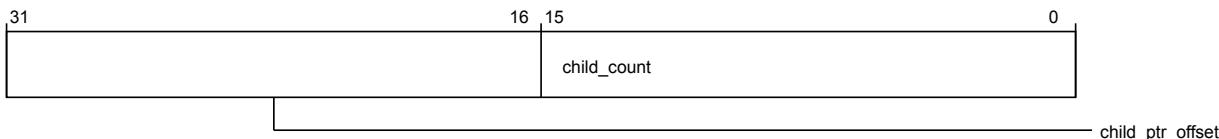


Figure 3-38 por_ccg_ha_por_ccg_ha_child_info (low)

The following table shows the por_ccg_ha_child_info lower register bit assignments.

Table 3-58 por_ccg_ha_por_ccg_ha_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'h0

por_ccg_ha_cfg_ctl

Functions as the configuration control register. Specifies the current mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccg_ha_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

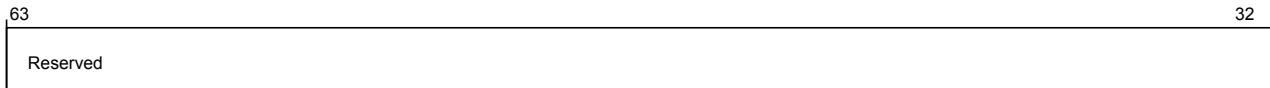


Figure 3-39 por_ccg_ha_por_ccg_ha_cfg_ctl (high)

The following table shows the por_ccg_ha_cfg_ctl higher register bit assignments.

Table 3-59 por_ccg_ha_por_ccg_ha_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

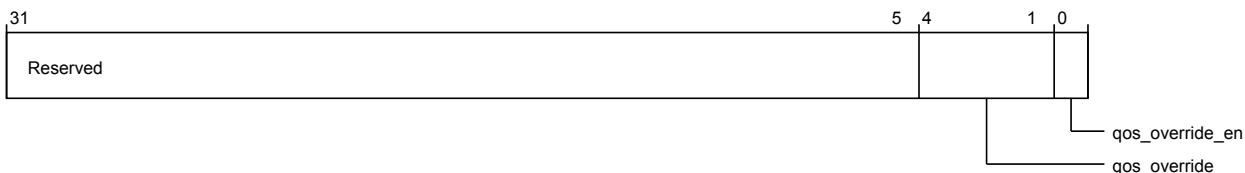


Figure 3-40 por_ccg_ha_por_ccg_ha_cfg_ctl (low)

The following table shows the por_ccg_ha_cfg_ctl lower register bit assignments.

Table 3-60 por_ccg_ha_por_ccg_ha_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:5	Reserved	Reserved	RO	-
4:1	qos_override	QoS override value	RW	4'b0
0	qos_override_en	QoS override en When set, QoS value on CHI side is driven from QoS override value in this register	RW	1'b0

por_ccg_ha_aux_ctl

Functions as the auxiliary control register for CXHA.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA08

Register reset 64'b0000001100001000

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-41 por_ccg_ha_por_ccg_ha_aux_ctl (high)

The following table shows the por_ccg_ha_aux_ctl higher register bit assignments.

Table 3-61 por_ccg_ha_por_ccg_ha_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

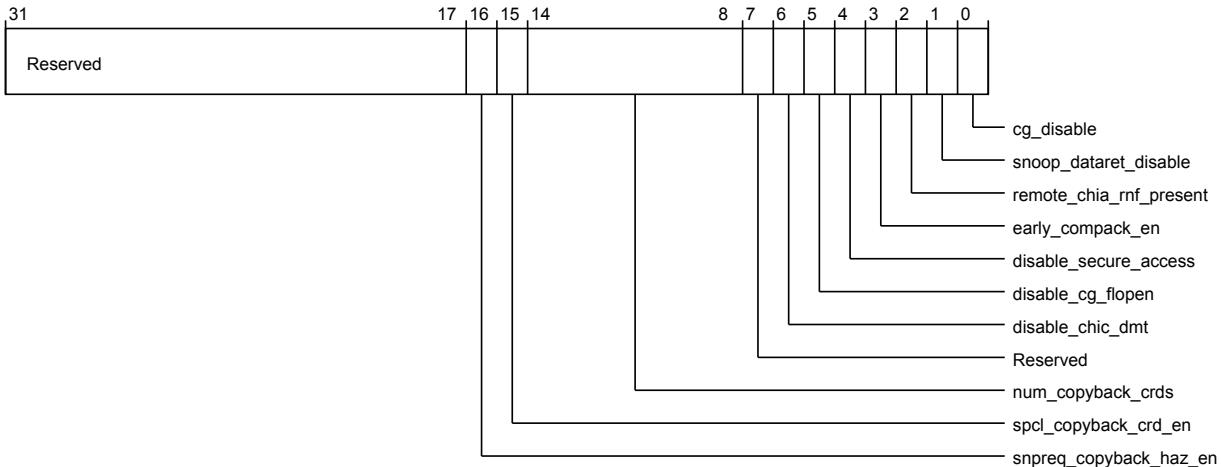


Figure 3-42 por_ccg_ha_por_ccg_ha_aux_ctl (low)

The following table shows the por_ccg_ha_aux_ctl lower register bit assignments.

Table 3-62 por_ccg_ha_por_ccg_ha_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	snpreq_copyback_haz_en	When set enables Snoop-CopyBack hazarding at incoming Snoop Request pipe	RW	1'b0
15	spcl_copyback_crd_en	When set enables sending special credits (from configured num_copyback_crds) for CopyBack Requests across all links	RW	1'b0
14:8	num_copyback_crds	Controls the total number of Request/Data credits reserved for CopyBack Requests across all links Note: This should be less than the Write Data Buffer depth	RW	7'h30
7	Reserved	Reserved	RO	-
6	disable_chic_dmt	When set disables CHI-C style DMT	RW	1'b0

Table 3-62 por_ccg_ha_por_ccg_ha_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
5	disable_cg_flopen	Disables enhanced flop enable control for dynamic power savings	RW	1'b0
4	disable_secure_access	Converts all accesses to non-secure	RW	1'b0
3	early_compack_en	Early CompAck enable; enables sending early CompAck on CCIX for requests that require CompAck	RW	1'b1
2	remote_chia_rnf_present	Indicates existence of CHIA RN-F in system; HA uses this indication to send SnpToS or SnpToSC 1'b0: HA converts SnpShared, SnpClean, and SnpNotSharedDirty to SnpToSC 1'b1: HA converts SnpShared, SnpClean, and SnpNotSharedDirty to SnpToS	RW	1'b0
1	snoop_dataret_disable	Disables setting data return for CCIX snoop requests for all CHI snoop opcodes	RW	1'b0
0	cg_disable	Disables clock gating when set	RW	1'b0

por_ccg_ha_mpam_control_link0

Controls MPAM override values on incoming CCIX Request in non-SMP mode for Link0

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA10
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccg_ha_secure_register_groups_override.mpam_ctl

The following image shows the higher register bit assignments.

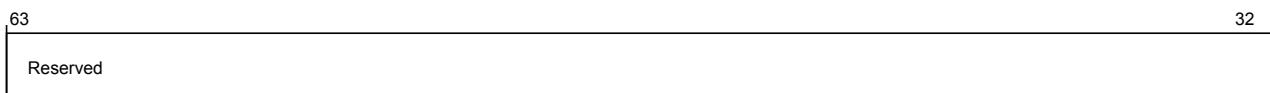


Figure 3-43 por_ccg_ha_por_ccg_ha_mpam_control_link0 (high)

The following table shows the por_ccg_ha_mpam_control_link0 higher register bit assignments.

Table 3-63 por_ccg_ha_por_ccg_ha_mpam_control_link0 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

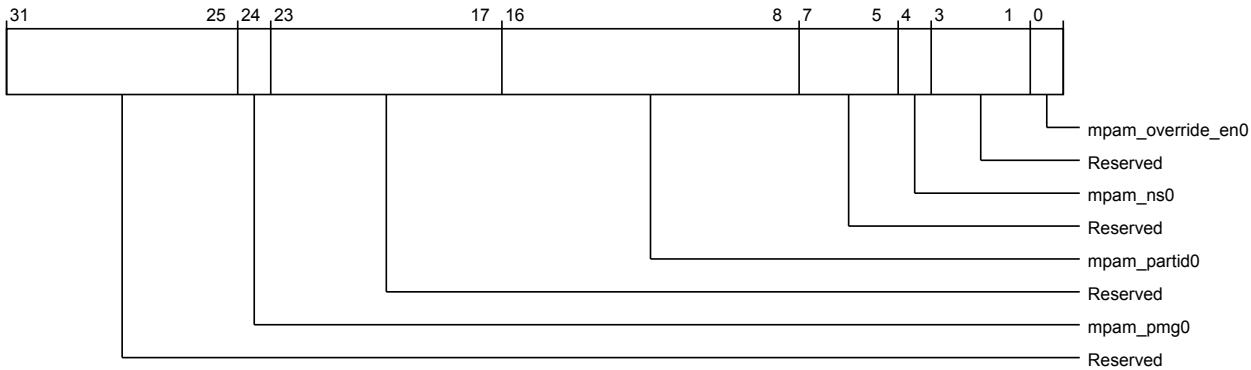


Figure 3-44 por_ccg_ha_por_ccg_ha_mpam_control_link0 (low)

The following table shows the por_ccg_ha_mpam_control_link0 lower register bit assignments.

Table 3-64 por_ccg_ha_por_ccg_ha_mpam_control_link0 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	mpam_pmg0	MPAM_PMG value	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	mpam_partid0	MPAM_PARTID value	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	mpam_ns0	MPAM_NS value	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	mpam_override_en0	MPAM override en When set, MPAM value on CHI side is driven from MPAM override value in this register. Applicable only in non-SMP mode	RW	1'b0

por_ccg_ha_mpam_control_link1

Controls MPAM override values on incoming CCIX Request in non-SMP mode for Link1

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA18
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccg_ha_secure_register_groups_override.mpam_ctl

The following image shows the higher register bit assignments.

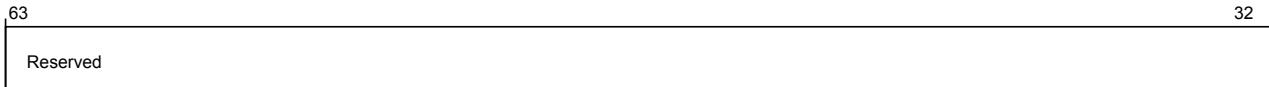


Figure 3-45 por_ccg_ha_por_ccg_ha_mpam_control_link1 (high)

The following table shows the por_ccg_ha_mpam_control_link1 higher register bit assignments.

Table 3-65 por_ccg_ha_por_ccg_ha_mpam_control_link1 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

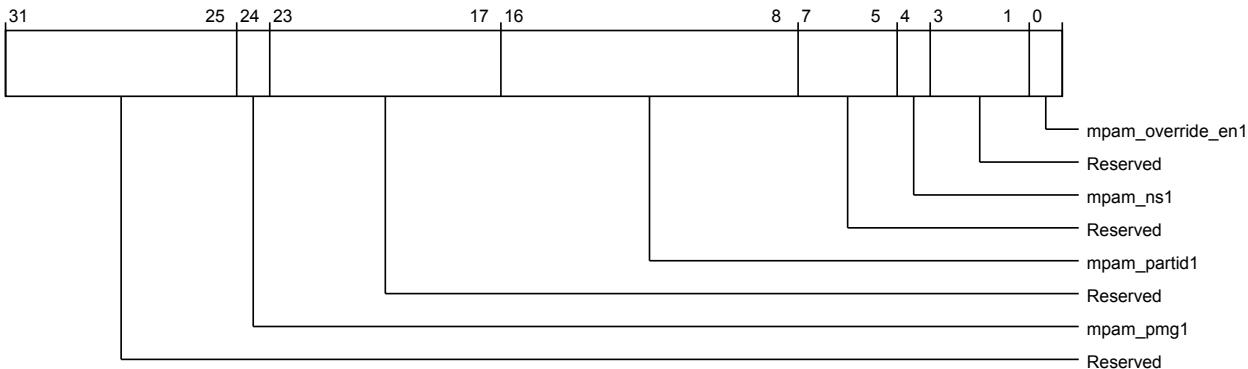


Figure 3-46 por_ccg_ha_por_ccg_ha_mpam_control_link1 (low)

The following table shows the por_ccg_ha_mpam_control_link1 lower register bit assignments.

Table 3-66 por_ccg_ha_por_ccg_ha_mpam_control_link1 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	mpam_pmg1	MPAM_PMG value	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	mpam_partid1	MPAM_PARTID value	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	mpam_ns1	MPAM_NS value	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	mpam_override_en1	MPAM override en When set, MPAM value on CHI side is driven from MPAM override value in this register. Applicable only in non-SMP mode	RW	1'b0

por_ccg_ha_mpam_control_link2

Controls MPAM override values on incoming CCIX Request in non-SMP mode for Link2

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA20
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccg_ha_secure_register_groups_override.mpam_ctl

The following image shows the higher register bit assignments.

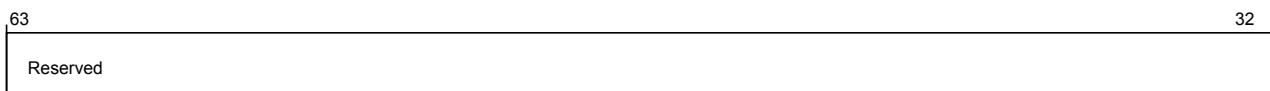


Figure 3-47 por_ccg_ha_por_ccg_ha_mpam_control_link2 (high)

The following table shows the por_ccg_ha_mpam_control_link2 higher register bit assignments.

Table 3-67 por_ccg_ha_por_ccg_ha_mpam_control_link2 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

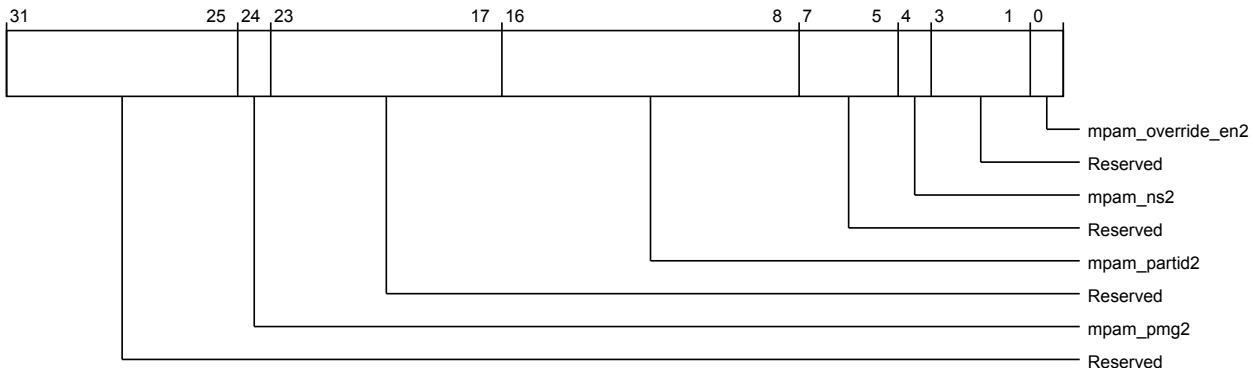


Figure 3-48 por_ccg_ha_por_ccg_ha_mpam_control_link2 (low)

The following table shows the por_ccg_ha_mpam_control_link2 lower register bit assignments.

Table 3-68 por_ccg_ha_por_ccg_ha_mpam_control_link2 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	mpam_pmg2	MPAM_PMG value	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	mpam_partid2	MPAM_PARTID value	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	mpam_ns2	MPAM_NS value	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	mpam_override_en2	MPAM override en When set, MPAM value on CHI side is driven from MPAM override value in this register. Applicable only in non-SMP mode	RW	1'b0

por_ccg_ha_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-49 por_ccg_ha_por_ccg_ha_secure_register_groups_override (high)

The following table shows the por_ccg_ha_secure_register_groups_override higher register bit assignments.

Table 3-69 por_ccg_ha_por_ccg_ha_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

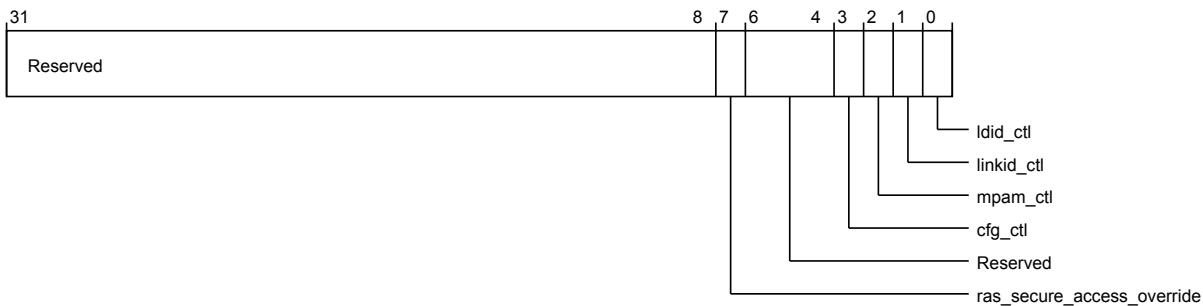


Figure 3-50 por_ccg_ha_por_ccg_ha_secure_register_groups_override (low)

The following table shows the por_ccg_ha_secure_register_groups_override lower register bit assignments.

Table 3-70 por_ccg_ha_por_ccg_ha_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0
6:4	Reserved	Reserved	RO	-
3	cfg_ctl	Allows non-secure access to secure HA config control registers	RW	1'b0
2	mpam_ctl	Allows non-secure access to secure HA MPAM override registers	RW	1'b0
1	linkid_ctl	Allows non-secure access to secure HA Link ID registers	RW	1'b0
0	ldid_ctl	Allows non-secure access to secure HA LDID registers	RW	1'b0

por_ccg_ha_unit_info

Provides component identification information for CXHA.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

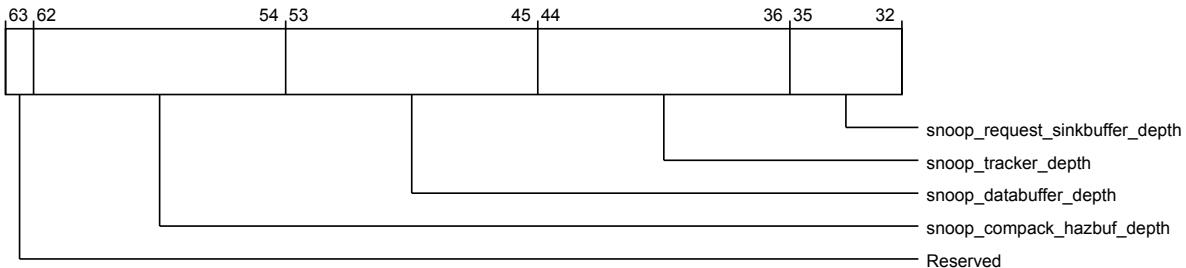


Figure 3-51 por_ccg_ha_por_ccg_ha_unit_info (high)

The following table shows the por_ccg_ha_unit_info higher register bit assignments.

Table 3-71 por_ccg_ha_por_ccg_ha_unit_info (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:54	snoop_compack_hazbuf_depth	Depth of CompAck snoop hazard buffer	RO	Configuration dependent
53:45	snoop_databuffer_depth	Depth of snoop data buffer	RO	Configuration dependent
44:36	snoop_tracker_depth	Depth of snoop tracker; number of outstanding SNP requests on CCIX	RO	Configuration dependent
35:32	snoop_request_sinkbuffer_depth	Depth of snoop request sink buffer; number of CHI SNP requests that can be sunk by CXHA	RO	Configuration dependent

The following image shows the lower register bit assignments.

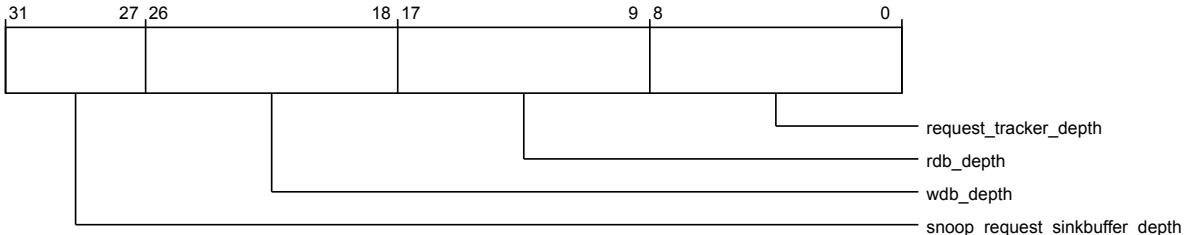


Figure 3-52 por_ccg_ha_por_ccg_ha_unit_info (low)

The following table shows the por_ccg_ha_unit_info lower register bit assignments.

Table 3-72 por_ccg_ha_por_ccg_ha_unit_info (low)

Bits	Field name	Description	Type	Reset
31:27	snoop_request_sinkbuffer_depth	Depth of snoop request sink buffer; number of CHI SNP requests that can be sunk by CXHA	RO	Configuration dependent
26:18	wdb_depth	Depth of write data buffer	RO	Configuration dependent
17:9	rdb_depth	Depth of read data buffer	RO	Configuration dependent
8:0	request_tracker_depth	Depth of request tracker	RO	Configuration dependent

por_ccg_ha_unit_info2

Provides additional component identification information for CXHA.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h908
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

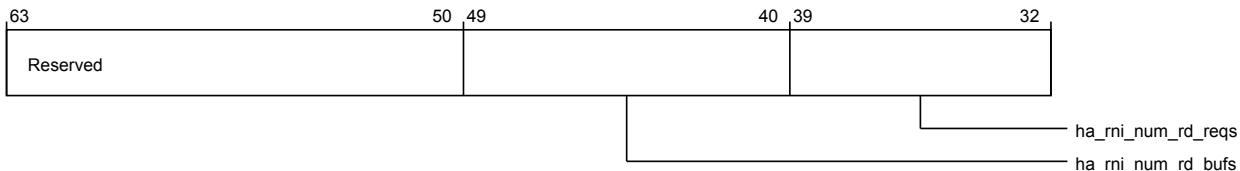


Figure 3-53 por_ccg_ha_por_ccg_ha_unit_info2 (high)

The following table shows the por_ccg_ha_unit_info2 higher register bit assignments.

Table 3-73 por_ccg_ha_por_ccg_ha_unit_info2 (high)

Bits	Field name	Description	Type	Reset
63:50	Reserved	Reserved	RO	-
49:40	ha_rni_num_rd_bufs	Number of HA_RNI read data buffers	RO	Configuration dependent
39:32	ha_rni_num_rd_reqs	Number of HA_RNI outstanding read requests	RO	Configuration dependent

The following image shows the lower register bit assignments.

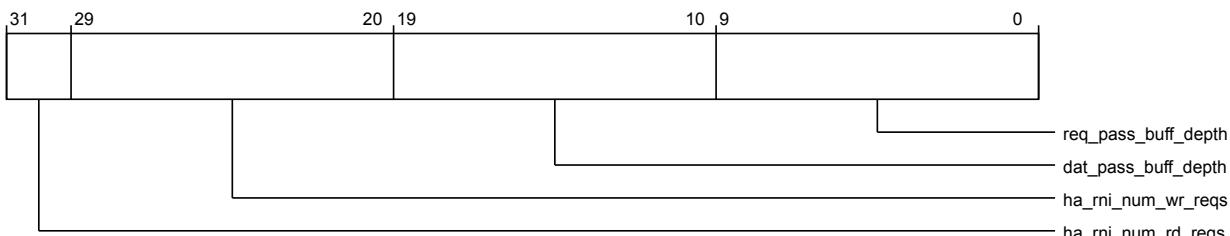


Figure 3-54 por_ccg_ha_por_ccg_ha_unit_info2 (low)

The following table shows the por_ccg_ha_unit_info2 lower register bit assignments.

Table 3-74 por_ccg_ha_por_ccg_ha_unit_info2 (low)

Bits	Field name	Description	Type	Reset
31:30	ha_rni_num_rd_reqs	Number of HA_RNI outstanding read requests	RO	Configuration dependent
29:20	ha_rni_num_wr_reqs	Number of HA_RNI outstanding write requests	RO	Configuration dependent

Table 3-74 por_ccg_ha_por_ccg_ha_unit_info2 (low) (continued)

Bits	Field name	Description	Type	Reset
19:10	dat_pass_buff_depth	Depth of DAT Passive Buffer	RO	Configuration dependent
9:0	req_pass_buff_depth	Depth of REQ Passive Buffer	RO	Configuration dependent

por_ccg_ha_agentid_to_linkid_reg0

Specifies the mapping of Agent ID to Link ID for Agent IDs 0 to 7.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F00

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

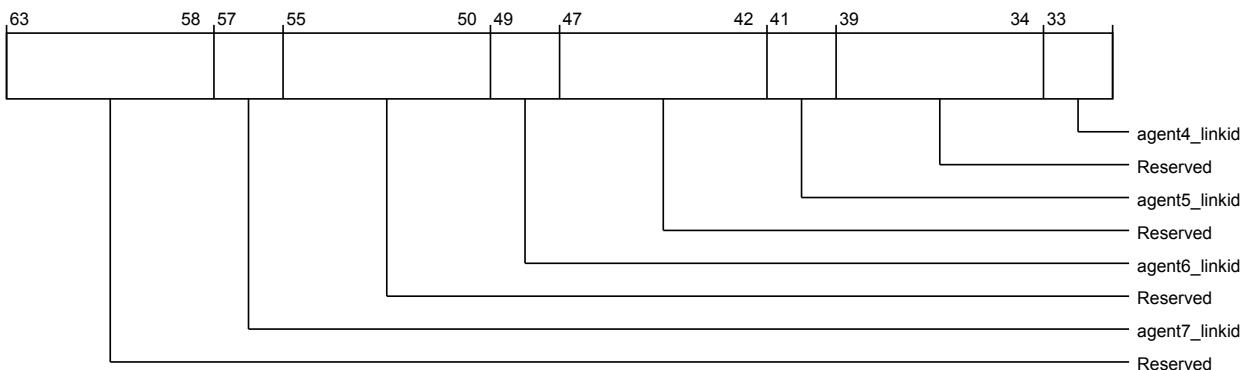


Figure 3-55 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg0 (high)

The following table shows the por_ccg_ha_agentid_to_linkid_reg0 higher register bit assignments.

Table 3-75 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg0 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent7_linkid	Specifies Link ID 7	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent6_linkid	Specifies Link ID 6	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent5_linkid	Specifies Link ID 5	RW	2'h0

Table 3-75 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg0 (high) (continued)

Bits	Field name	Description	Type	Reset
39:34	Reserved	Reserved	RO	-
33:32	agent4_linkid	Specifies Link ID 4	RW	2'h0

The following image shows the lower register bit assignments.

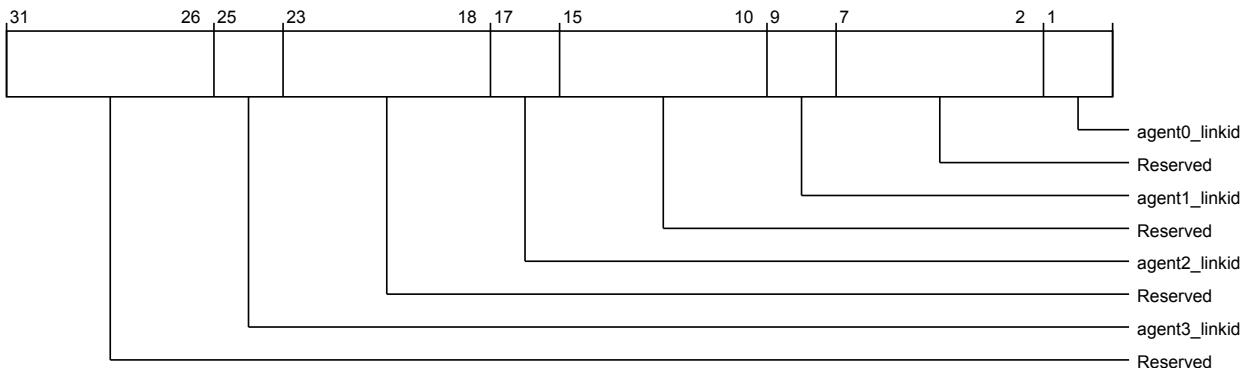


Figure 3-56 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg0 (low)

The following table shows the por_ccg_ha_agentid_to_linkid_reg0 lower register bit assignments.

Table 3-76 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg0 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent3_linkid	Specifies Link ID 3	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent2_linkid	Specifies Link ID 2	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent1_linkid	Specifies Link ID 1	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent0_linkid	Specifies Link ID 0	RW	2'h0

por_ccg_ha_agentid_to_linkid_reg1

Specifies the mapping of Agent ID to Link ID for Agent IDs 8 to 15.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1F08
Register reset	64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

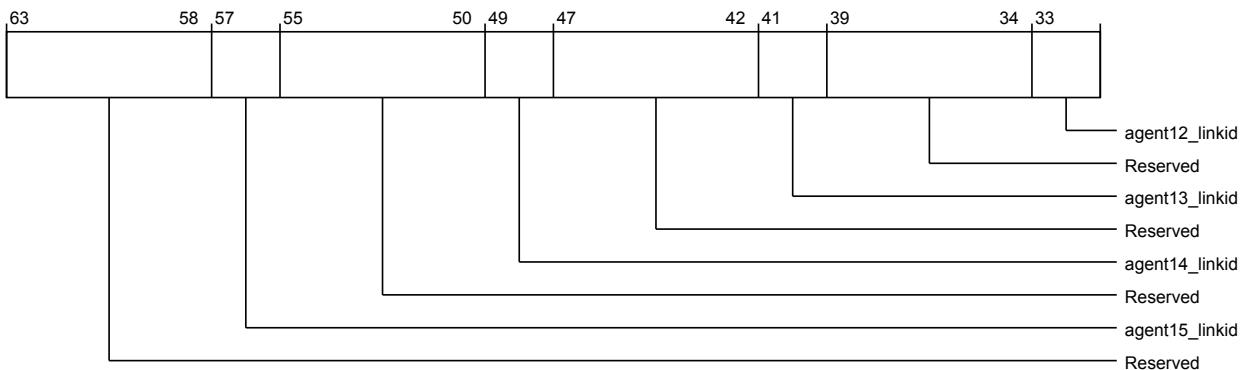


Figure 3-57 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg1 (high)

The following table shows the por_ccg_ha_agentid_to_linkid_reg1 higher register bit assignments.

Table 3-77 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg1 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent15_linkid	Specifies Link ID 15	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent14_linkid	Specifies Link ID 14	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent13_linkid	Specifies Link ID 13	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent12_linkid	Specifies Link ID 12	RW	2'h0

The following image shows the lower register bit assignments.

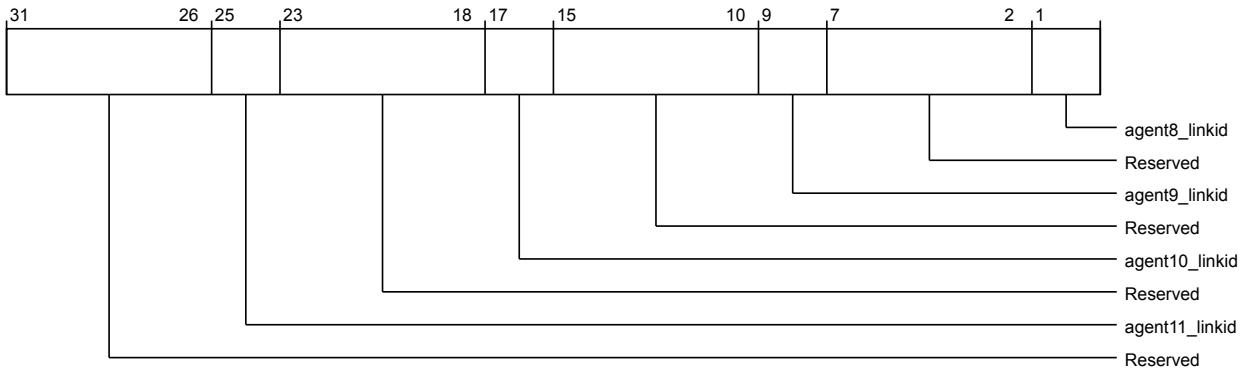


Figure 3-58 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg1 (low)

The following table shows the por_ccg_ha_agentid_to_linkid_reg1 lower register bit assignments.

Table 3-78 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg1 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent11_linkid	Specifies Link ID 11	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent10_linkid	Specifies Link ID 10	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent9_linkid	Specifies Link ID 9	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent8_linkid	Specifies Link ID 8	RW	2'h0

por_ccg_ha_agentid_to_linkid_reg2

Specifies the mapping of Agent ID to Link ID for Agent IDs 16 to 23.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F10

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

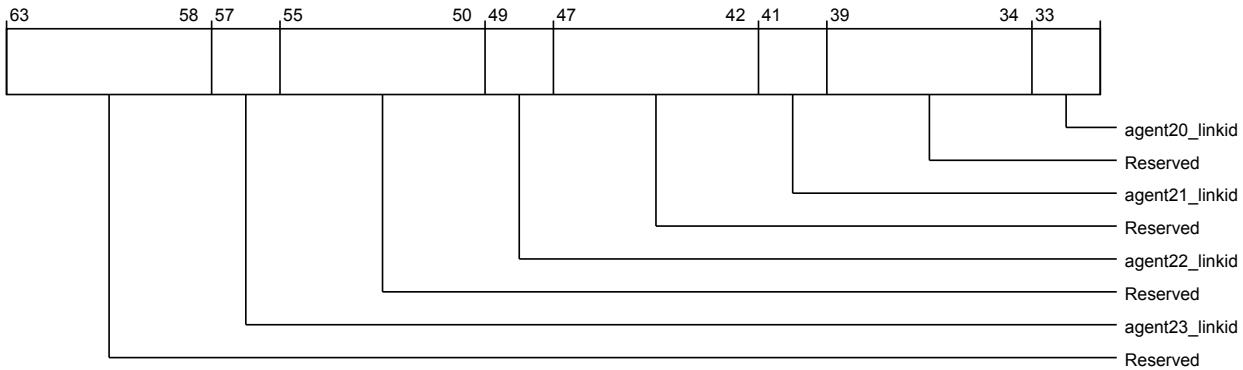


Figure 3-59 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg2 (high)

The following table shows the por_ccg_ha_agentid to linkid_reg2 higher register bit assignments.

Table 3-79 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg2 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent23_linkid	Specifies Link ID 23	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent22_linkid	Specifies Link ID 22	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent21_linkid	Specifies Link ID 21	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent20_linkid	Specifies Link ID 20	RW	2'h0

The following image shows the lower register bit assignments.

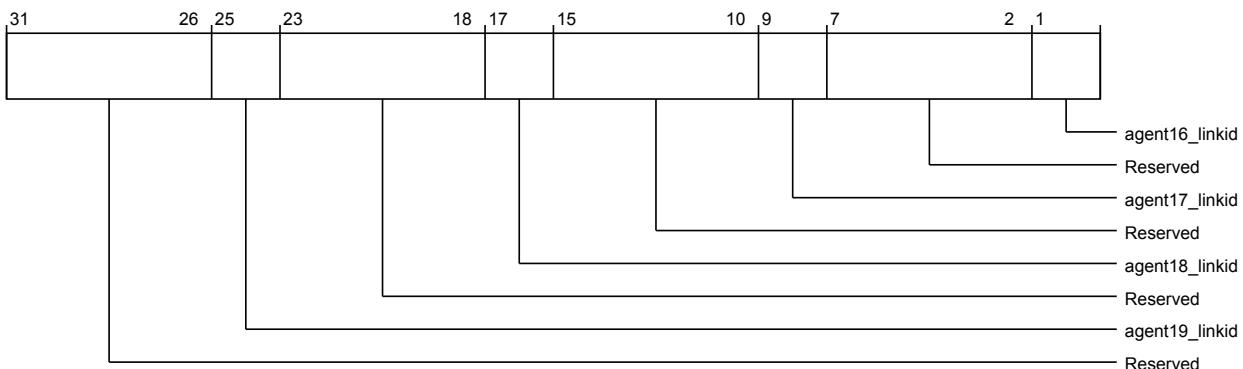


Figure 3-60 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg2 (low)

The following table shows the port assignments to linkid register bit assignments.

Table 3-80 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg2 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent19_linkid	Specifies Link ID 19	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent18_linkid	Specifies Link ID 18	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent17_linkid	Specifies Link ID 17	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent16_linkid	Specifies Link ID 16	RW	2'h0

por_ccg_ha_agentid_to_linkid_reg3

Specifies the mapping of Agent ID to Link ID for Agent IDs 24 to 31.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F18

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

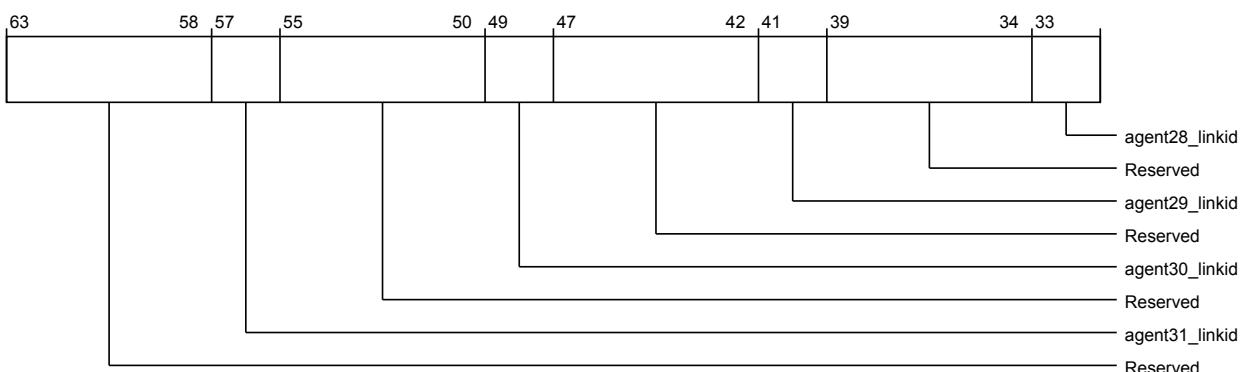


Figure 3-61 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg3 (high)

The following table shows the por_ccg_ha_agentid_to_linkid_reg3 higher register bit assignments.

Table 3-81 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg3 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent31_linkid	Specifies Link ID 31	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent30_linkid	Specifies Link ID 30	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent29_linkid	Specifies Link ID 29	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent28_linkid	Specifies Link ID 28	RW	2'h0

The following image shows the lower register bit assignments.

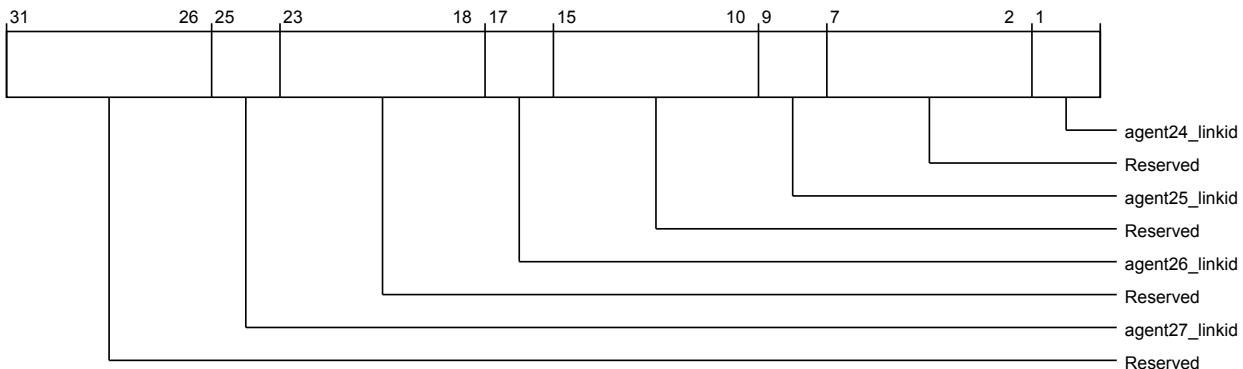


Figure 3-62 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg3 (low)

The following table shows the por_ccg_ha_agentid_to_linkid_reg3 lower register bit assignments.

Table 3-82 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg3 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent27_linkid	Specifies Link ID 27	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent26_linkid	Specifies Link ID 26	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent25_linkid	Specifies Link ID 25	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent24_linkid	Specifies Link ID 24	RW	2'h0

por_ccg_ha_agentid_to_linkid_reg4

Specifies the mapping of Agent ID to Link ID for Agent IDs 32 to 39.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1F20
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

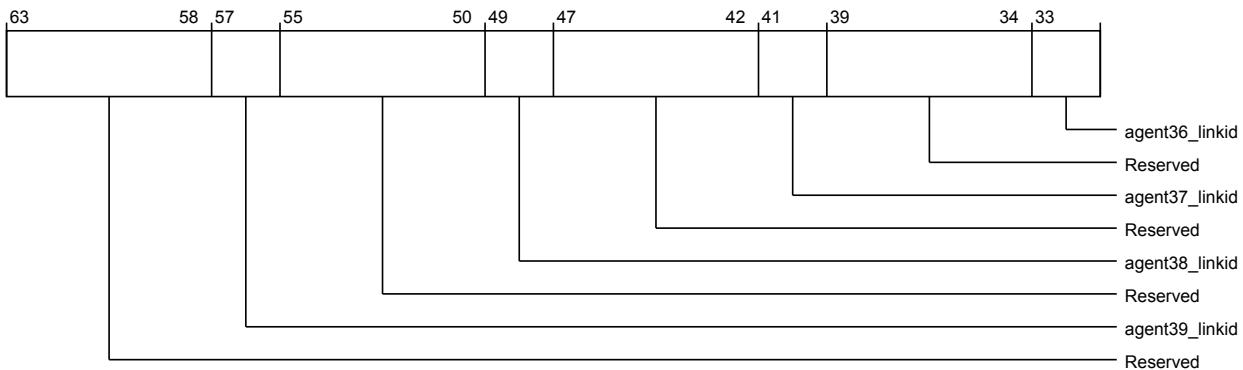


Figure 3-63 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg4 (high)

The following table shows the por_ccg_ha_agentid_to_linkid_reg4 higher register bit assignments.

Table 3-83 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg4 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent39_linkid	Specifies Link ID 39	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent38_linkid	Specifies Link ID 38	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent37_linkid	Specifies Link ID 37	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent36_linkid	Specifies Link ID 36	RW	2'h0

The following image shows the lower register bit assignments.

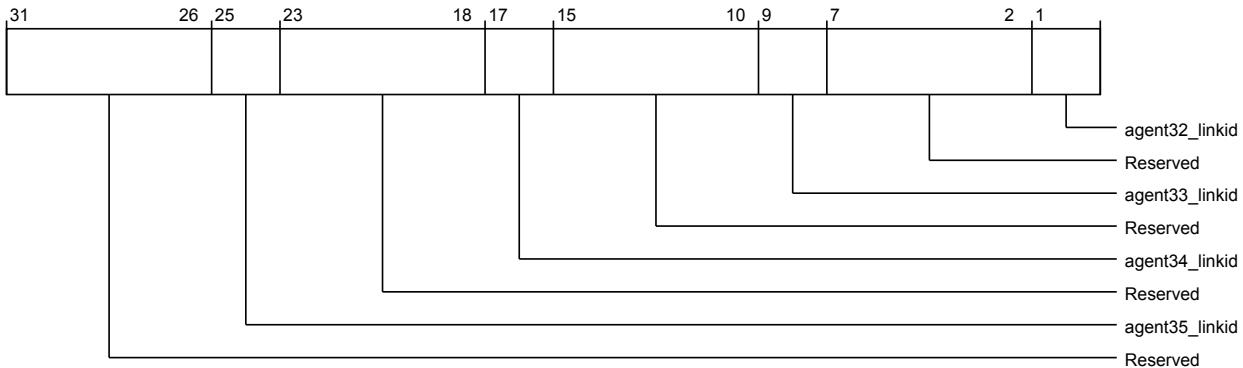


Figure 3-64 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg4 (low)

The following table shows the por_ccg_ha agentid to linkid reg4 lower register bit assignments.

Table 3-84 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg4 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent35_linkid	Specifies Link ID 35	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent34_linkid	Specifies Link ID 34	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent33_linkid	Specifies Link ID 33	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent32_linkid	Specifies Link ID 32	RW	2'h0

por_ccg_ha_agentid_to_linkid_reg5

Specifies the mapping of Agent ID to Link ID for Agent IDs 40 to 47.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_ccg_ha_secure_register_groups_override.linkid_ctl

The following is a code example which illustrates this issue:

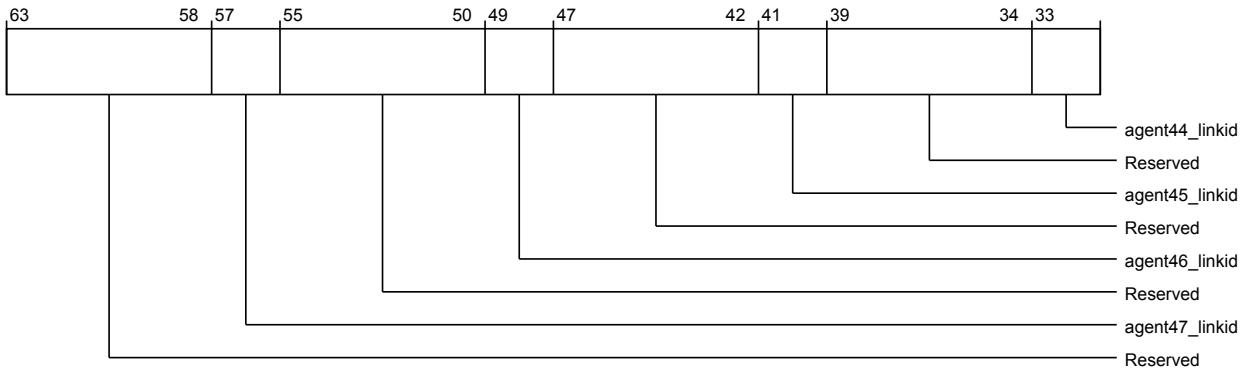


Figure 3-65 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg5 (high)

The following table shows the por_ccg_ha_agentid_to_linkid_reg5 higher register bit assignments.

Table 3-85 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg5 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent47_linkid	Specifies Link ID 47	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent46_linkid	Specifies Link ID 46	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent45_linkid	Specifies Link ID 45	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent44_linkid	Specifies Link ID 44	RW	2'h0

The following image shows the lower register bit assignments.

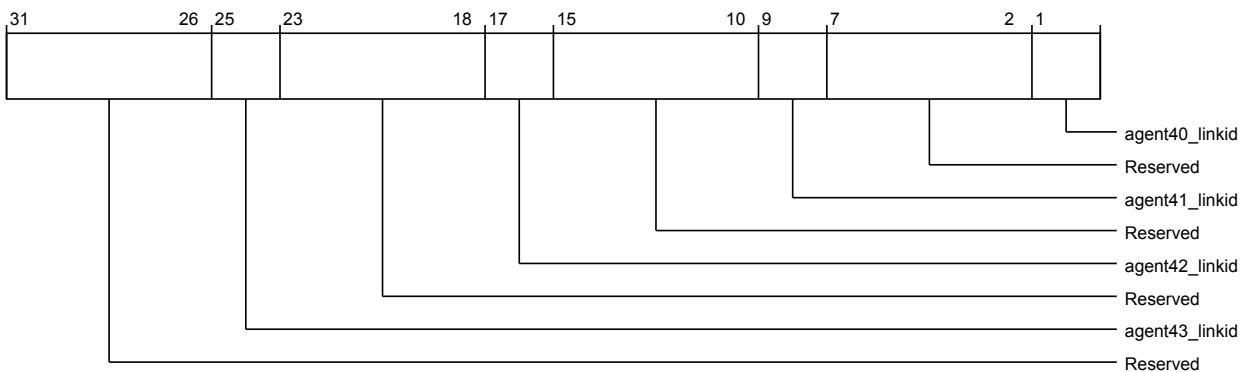


Figure 3-66 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg5 (low)

The following table shows the por_ccg_ha_agentid_to_linkid_reg5 lower register bit assignments.

Table 3-86 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg5 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent43_linkid	Specifies Link ID 43	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent42_linkid	Specifies Link ID 42	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent41_linkid	Specifies Link ID 41	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent40_linkid	Specifies Link ID 40	RW	2'h0

por_ccg_ha_agentid_to_linkid_reg6

Specifies the mapping of Agent ID to Link ID for Agent IDs 48 to 55.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F30

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

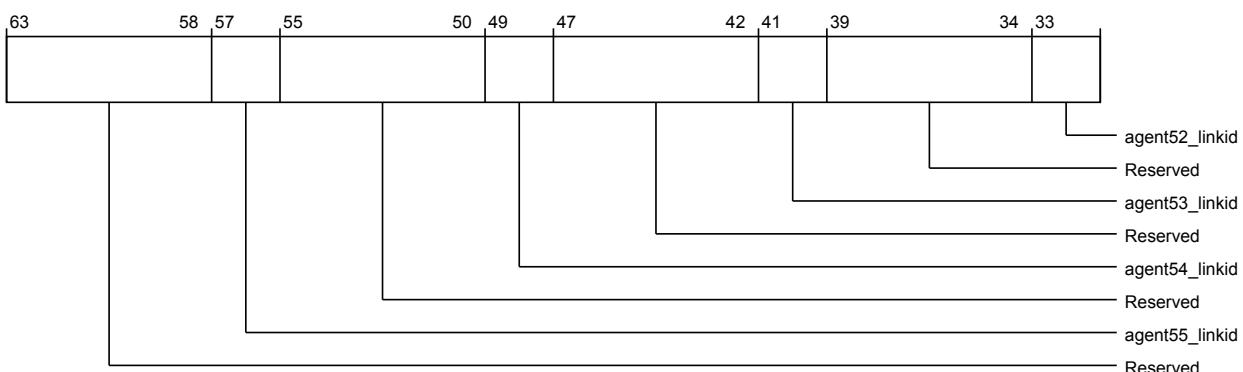


Figure 3-67 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg6 (high)

The following table shows the por_ccg_ha_agentid_to_linkid_reg6 higher register bit assignments.

Table 3-87 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg6 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent55_linkid	Specifies Link ID 55	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent54_linkid	Specifies Link ID 54	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent53_linkid	Specifies Link ID 53	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent52_linkid	Specifies Link ID 52	RW	2'h0

The following image shows the lower register bit assignments.

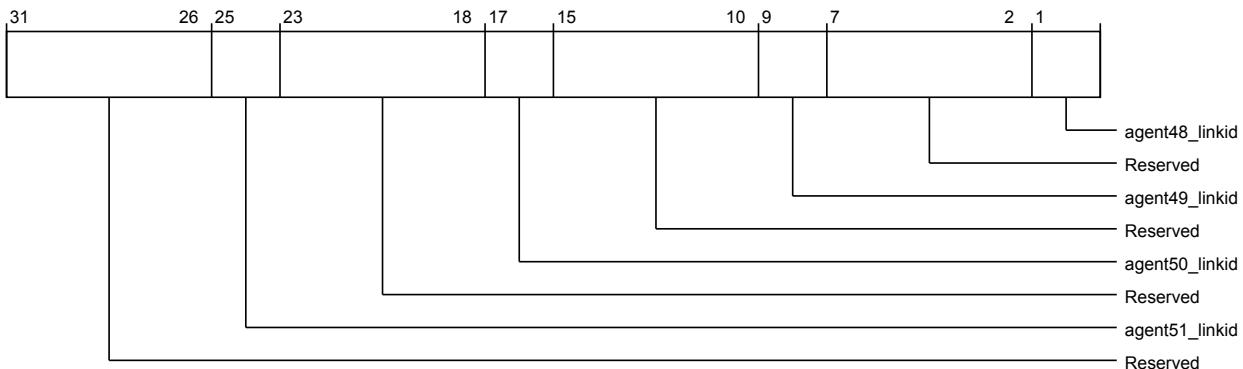


Figure 3-68 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg6 (low)

The following table shows the por_ccg_ha_agentid_to_linkid_reg6 lower register bit assignments.

Table 3-88 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg6 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent51_linkid	Specifies Link ID 51	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent50_linkid	Specifies Link ID 50	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent49_linkid	Specifies Link ID 49	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent48_linkid	Specifies Link ID 48	RW	2'h0

por_ccg_ha_agentid_to_linkid_reg7

Specifies the mapping of Agent ID to Link ID for Agent IDs 56 to 63.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1F38
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

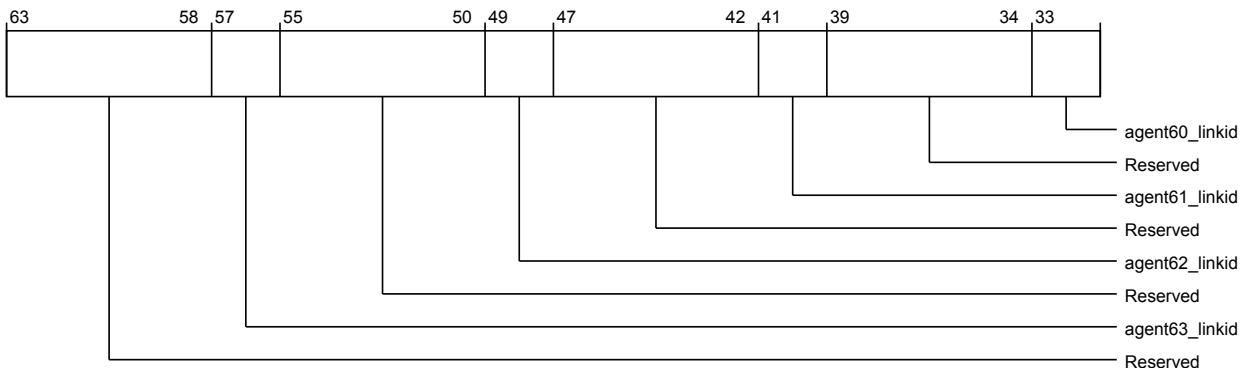


Figure 3-69 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg7 (high)

The following table shows the port configuration assignments for the higher register bit assignments.

Table 3-89 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg7 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent63_linkid	Specifies Link ID 63	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent62_linkid	Specifies Link ID 62	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent61_linkid	Specifies Link ID 61	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent60_linkid	Specifies Link ID 60	RW	2'h0

The following image shows the lower register bit assignments.

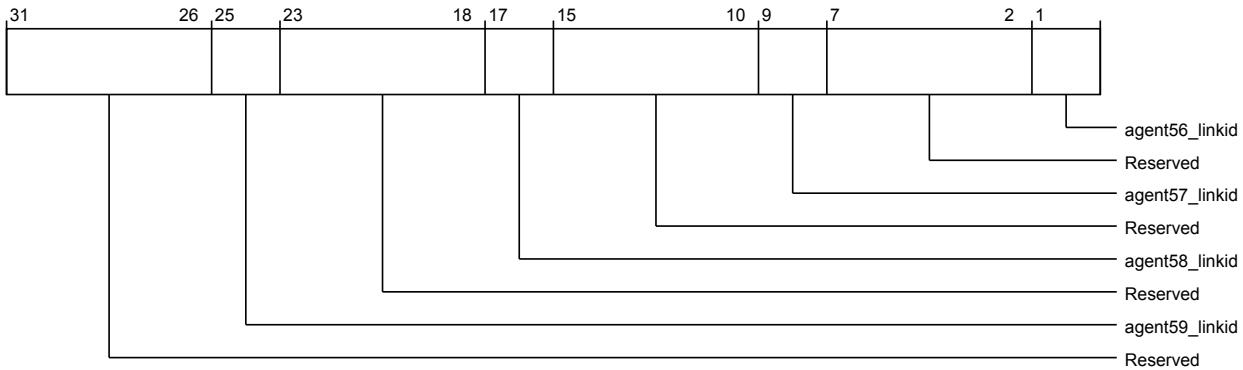


Figure 3-70 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg7 (low)

The following table shows the por_ccg_ha_agentid_to_linkid_reg7 lower register bit assignments.

Table 3-90 por_ccg_ha_por_ccg_ha_agentid_to_linkid_reg7 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent59_linkid	Specifies Link ID 59	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent58_linkid	Specifies Link ID 58	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent57_linkid	Specifies Link ID 57	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent56_linkid	Specifies Link ID 56	RW	2'h0

por_ccg_ha_agentid_to_linkid_val

Specifies which Agent ID to Link ID mappings are valid.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1FF8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

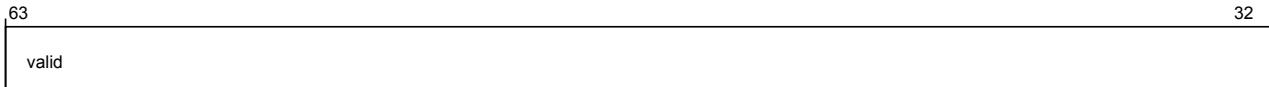


Figure 3-71 por_ccg_ha_por_ccg_ha_agentid_to_linkid_val (high)

The following table shows the por_ccg_ha_agentid_to_linkid_val higher register bit assignments.

Table 3-91 por_ccg_ha_por_ccg_ha_agentid_to_linkid_val (high)

Bits	Field name	Description	Type	Reset
63:32	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

The following image shows the lower register bit assignments.

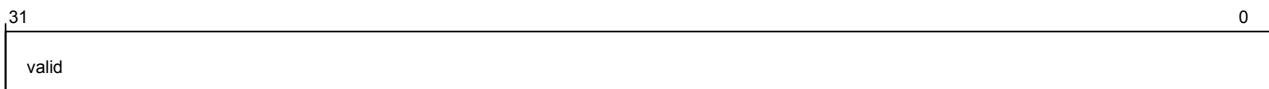


Figure 3-72 por_ccg_ha_por_ccg_ha_agentid_to_linkid_val (low)

The following table shows the por_ccg_ha_agentid_to_linkid_val lower register bit assignments.

Table 3-92 por_ccg_ha_por_ccg_ha_agentid_to_linkid_val (low)

Bits	Field name	Description	Type	Reset
31:0	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255

This register repeats 255 times. It parametrized by the index from 0 to 255. Specifies the mapping of Expanded RAID to RN-F LDID for Expanded RAIDs # $\{index*4\}$ to # $\{index*4+3\}$.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + # $\{8*[0, 1, 2, \dots, 254, 255]\}$
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ha_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

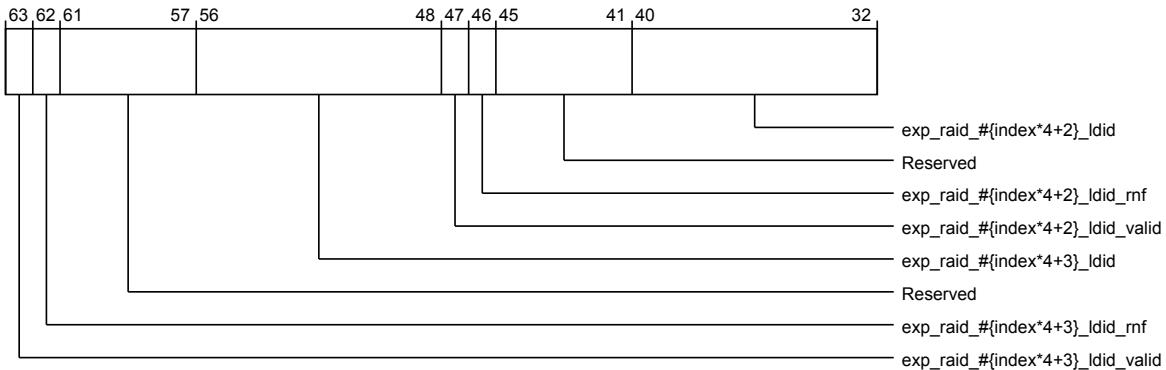


Figure 3-73 por_ccg_ha_por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255 (high)

The following table shows the por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255 higher register bit assignments.

Table 3-93 por_ccg_ha_por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255 (high)

Bits	Field name	Description	Type	Reset
63	exp_raid_{index*4+3}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4+3} is valid	RW	1'b0
62	exp_raid_{index*4+3}_ldid_mf	Specifies if Expanded RAID #{index*4+3} is RN-F	RW	1'b0
61:57	Reserved	Reserved	RO	-
56:48	exp_raid_{index*4+3}_ldid	Specifies the LDID for Expanded RAID #{index*4+3}	RW	9'h0
47	exp_raid_{index*4+2}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4+2} is valid	RW	1'b0
46	exp_raid_{index*4+2}_ldid_mf	Specifies if Expanded RAID #{index*4+2} is RN-F	RW	1'b0
45:41	Reserved	Reserved	RO	-
40:32	exp_raid_{index*4+2}_ldid	Specifies the LDID for Expanded RAID #{index*4+2}	RW	9'h0

The following image shows the lower register bit assignments.

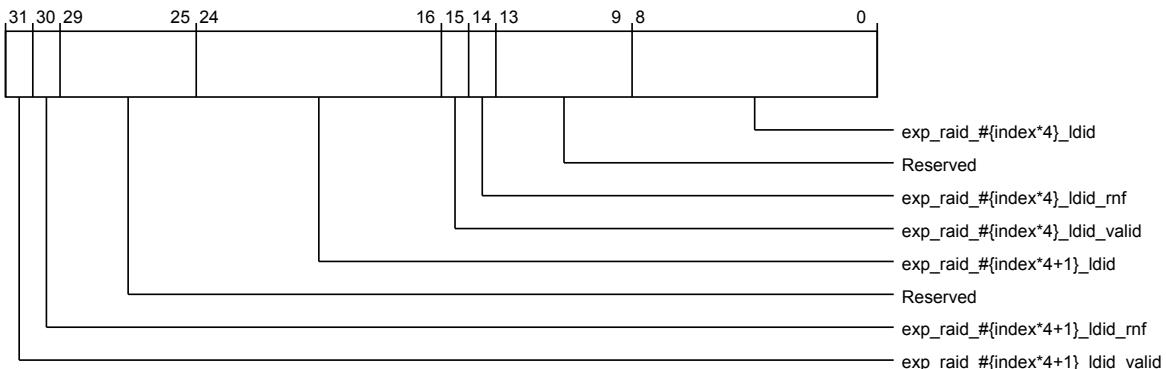


Figure 3-74 por_ccg_ha_por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255 (low)

The following table shows the por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255 lower register bit assignments.

Table 3-94 por_ccg_ha_por_ccg_ha_rnf_exp_raid_to_ldid_reg_0-255 (low)

Bits	Field name	Description	Type	Reset
31	exp_raid_#{index*4+1}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4+1} is valid	RW	1'b0
30	exp_raid_#{index*4+1}_ldid_rnf	Specifies if Expanded RAID #{index*4+1} is RN-F	RW	1'b0
29:25	Reserved	Reserved	RO	-
24:16	exp_raid_#{index*4+1}_ldid	Specifies the LDID for Expanded RAID #{index*4+1}	RW	9'h0
15	exp_raid_#{index*4}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4} is valid	RW	1'b0
14	exp_raid_#{index*4}_ldid_rnf	Specifies if Expanded RAID #{index*4} is RN-F	RW	1'b0
13:9	Reserved	Reserved	RO	-
8:0	exp_raid_#{index*4}_ldid	Specifies the LDID for Expanded RAID #{index*4}	RW	9'h0

por_ccg_ha_pmu_event_sel

Specifies the PMU event to be counted as a 7-bit ID with the following encodings: 7'h00:
 CXHA_PMU_EVENT_NULL 7'h61: CXHA_PMU_EVENT_RDDATBYP 7'h62:
 CXHA_PMU_EVENT_CHIRSP_UP_STALL 7'h63: CXHA_PMU_EVENT_CHIDAT_UP_STALL
 7'h64: CXHA_PMU_EVENT_SNPPCRD_LNK0_STALL 7'h65:
 CXHA_PMU_EVENT_SNPPCRD_LNK1_STALL 7'h66:
 CXHA_PMU_EVENT_SNPPCRD_LNK2_STALL 7'h67: CXHA_PMU_EVENT_REQTRK_OCC
 7'h68: CXHA_PMU_EVENT_RDB_OCC 7'h69: CXHA_PMU_EVENT_RDBBYP_OCC 7'h6A:
 CXHA_PMU_EVENT_WDB_OCC 7'h6B: CXHA_PMU_EVENT_SNPTRK_OCC 7'h6C:
 CXHA_PMU_EVENT_SDB_OCC 7'h6D: CXHA_PMU_EVENT_SNPHAZ_OCC 7'h6E:
 CXHA_PMU_EVENT_REQTRK_ALLOC 7'h6F: CXHA_PMU_EVENT_RDB_ALLOC 7'h70:
 CXHA_PMU_EVENT_RDBBYP_ALLOC 7'h71: CXHA_PMU_EVENT_WDB_ALLOC 7'h72:
 CXHA_PMU_EVENT_SNPTRK_ALLOC 7'h73: CXHA_PMU_EVENT_SDB_ALLOC 7'h74:
 CXHA_PMU_EVENT_SNPHAZ_ALLOC

Its characteristics are:

Type RW
Register width (Bits) 64
Address offset 16'h2000
Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

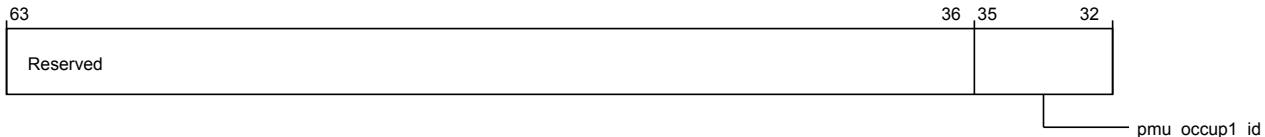


Figure 3-75 por_ccg_ha_por_ccg_ha_pmu_event_sel (high)

The following table shows the por_ccg_ha_pmu_event_sel higher register bit assignments.

Table 3-95 por_ccg_ha_por_ccg_ha_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:36	Reserved	Reserved	RO	-
35:32	pmu_occup1_id	CXHA PMU occupancy event selector ID	RW	4'b0

The following image shows the lower register bit assignments.

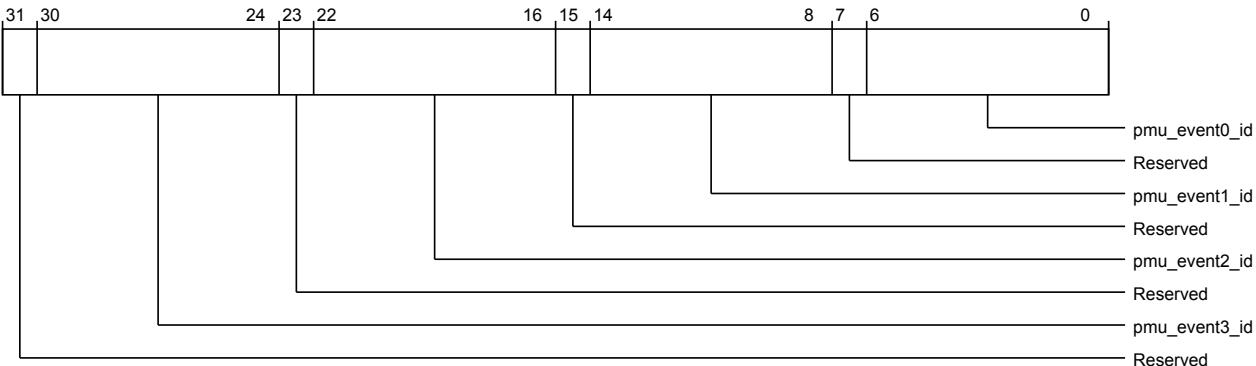


Figure 3-76 por_ccg_ha_por_ccg_ha_pmu_event_sel (low)

The following table shows the port configuration assignments for the PMU event selection register bits.

Table 3-96 por_ccg_ha_por_ccg_ha_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:24	pmu_event3_id	CXHA PMU Event 3 ID	RW	7'b0
23	Reserved	Reserved	RO	-
22:16	pmu_event2_id	CXHA PMU Event 2 ID	RW	7'b0
15	Reserved	Reserved	RO	-
14:8	pmu_event1_id	CXHA PMU Event 1 ID	RW	7'b0
7	Reserved	Reserved	RO	-
6:0	pmu_event0_id	CXHA PMU Event 0 ID	RW	7'b0

por_ccg ha cxprtcl_link0_ctl

Functions as the CXHA CCIX Protocol Link 0 control register. Works with por ccg ha cexprtcl link0 status.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1C00

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

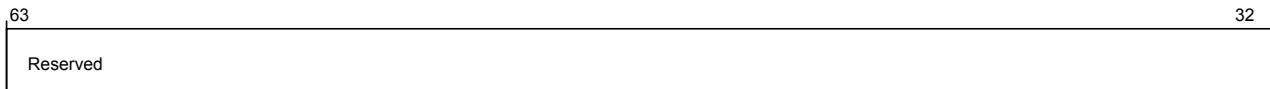


Figure 3-77 por_ccg_ha_por_ccg_ha_cxprtcl_link0_ctl (high)

The following table shows the por_ccg_ha_cxprtcl_link0_ctl higher register bit assignments.

Table 3-97 por_ccg_ha_por_ccg_ha_cxprtcl_link0_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

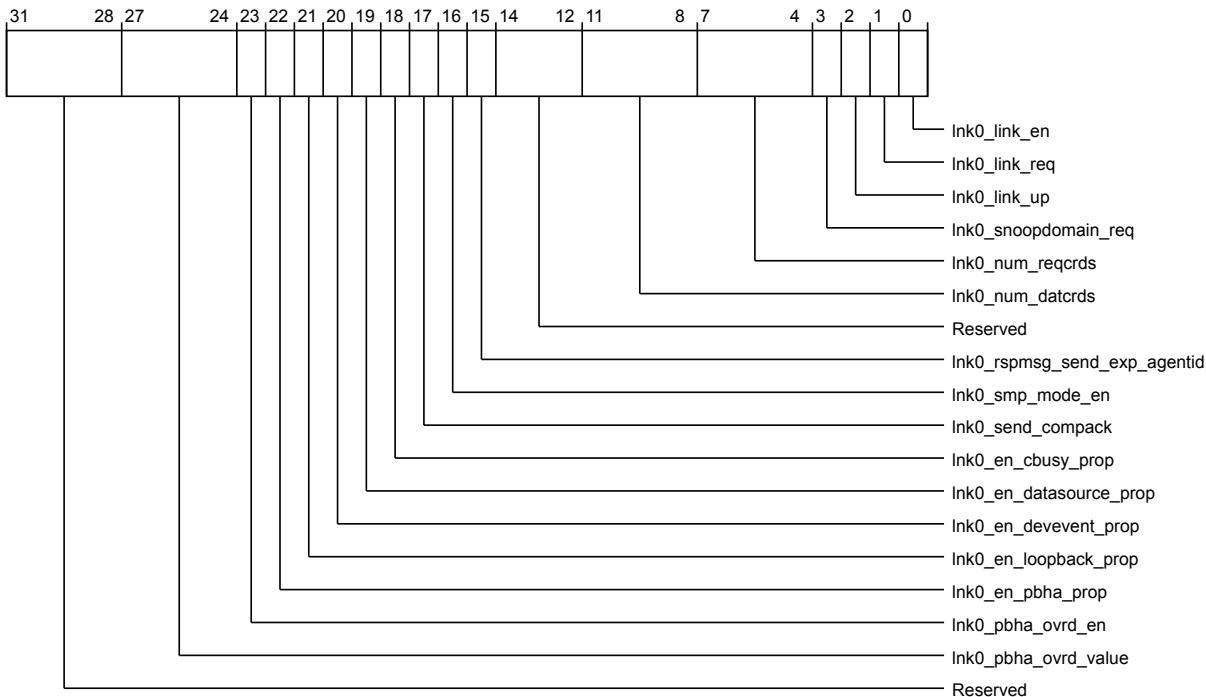


Figure 3-78 por_ccg_ha_por_ccg_ha_cxprtcl_link0_ctl (low)

The following table shows the por_ccg_ha_cxprtcl_link0_ctl lower register bit assignments.

Table 3-98 por_ccg_ha_por_ccg_ha_cxprtcl_link0_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	lnk0_pbha_ovrd_value	Override value for PBHA on CCIX Link 0. Applicable only when lnk0_pbha_ovrd_en is set. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	4'b0
23	lnk0_pbha_ovrd_en	When set, overrides PBHA on CCIX Link 0. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	1'b0
22	lnk0_en_pbha_prop	When set, enables propagation of PBHA on CCIX Link 0.	RW	1'b0
21	lnk0_en_loopback_prop	When set, enables propagation of LoopBack on CCIX Link 0.	RW	1'b0
20	lnk0_en_devevent_prop	When set, enables propagation of DevEvent on CCIX Link 0.	RW	1'b1
19	lnk0_en_datasource_prop	When set, enables propagation of DataSource on CCIX Link 0.	RW	1'b1
18	lnk0_en_cbusy_prop	When set, enables propagation of CBusy on CCIX Link 0.	RW	1'b1
17	lnk0_send_compack	When set, sends CompAck for CCIX Link 0.	RW	1'b0
16	lnk0_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 0.	RW	Configuration dependent
15	lnk0_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 0	RW	1'b0
14:12	Reserved	Reserved	RO	-
11:8	lnk0_num_datcrds	Controls the number of CCIX data credits assigned to Link 0 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
7:4	lnk0_num_reqcrds	Controls the number of CCIX request credits assigned to Link 0 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0

Table 3-98 por_ccg_ha_por_ccg_ha_cxprtcl_link0_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
3	lnk0_snoopdomain_req	Controls Snoop domain enable (SYSCOREQ) for CCIX Link 0	RW	1'b0
2	lnk0_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk0_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or</p> <p>Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk0_link_en	<p>Enables CCIX Link 0 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_ccg_ha_cxprtcl_link0_status

Functions as the CXHA CCIX Protocol Link 0 status register. Works with por_ccg_ha_cxprtcl_link0_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C08

Register reset 64'b010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

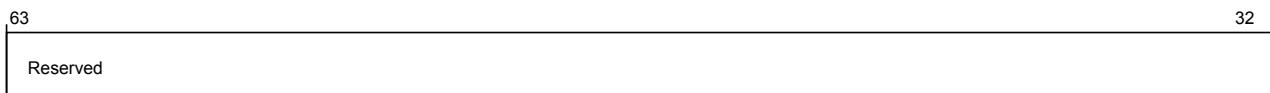


Figure 3-79 por_ccg_ha_por_ccg_ha_cxprtcl_link0_status (high)

The following table shows the por_ccg_ha_cxprtcl_link0_status higher register bit assignments.

Table 3-99 por_ccg_ha_por_ccg_ha_cxprtcl_link0_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

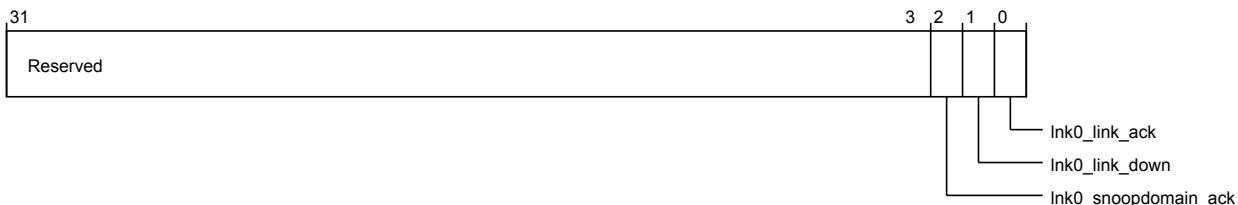


Figure 3-80 por_ccg_ha_por_ccg_ha_cxprtcl_link0_status (low)

The following table shows the por_ccg_ha_cxprtcl_link0_status lower register bit assignments.

Table 3-100 por_ccg_ha_por_ccg_ha_cxprtcl_link0_status (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	Lnk0_snoopdomain_ack	Provides Snoop domain status (SYSCOACK) for CCIX Link 0	RO	1'b0
1	Lnk0_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	Lnk0_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_ccg_ha_cxprtcl_link1_ctl

Functions as the CXHA CCIX Protocol Link 1 control register. Works with por_ccg_ha_cxprtcl_link1_status.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C10

Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

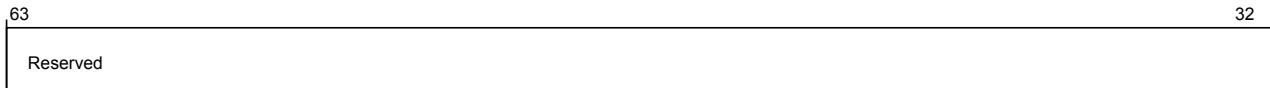


Figure 3-81 por_ccg_ha_por_ccg_ha_cxprtcl_link1_ctl (high)

The following table shows the por_ccg_ha_cxprtcl_link1_ctl higher register bit assignments.

Table 3-101 por_ccg_ha_por_ccg_ha_cxprtcl_link1_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

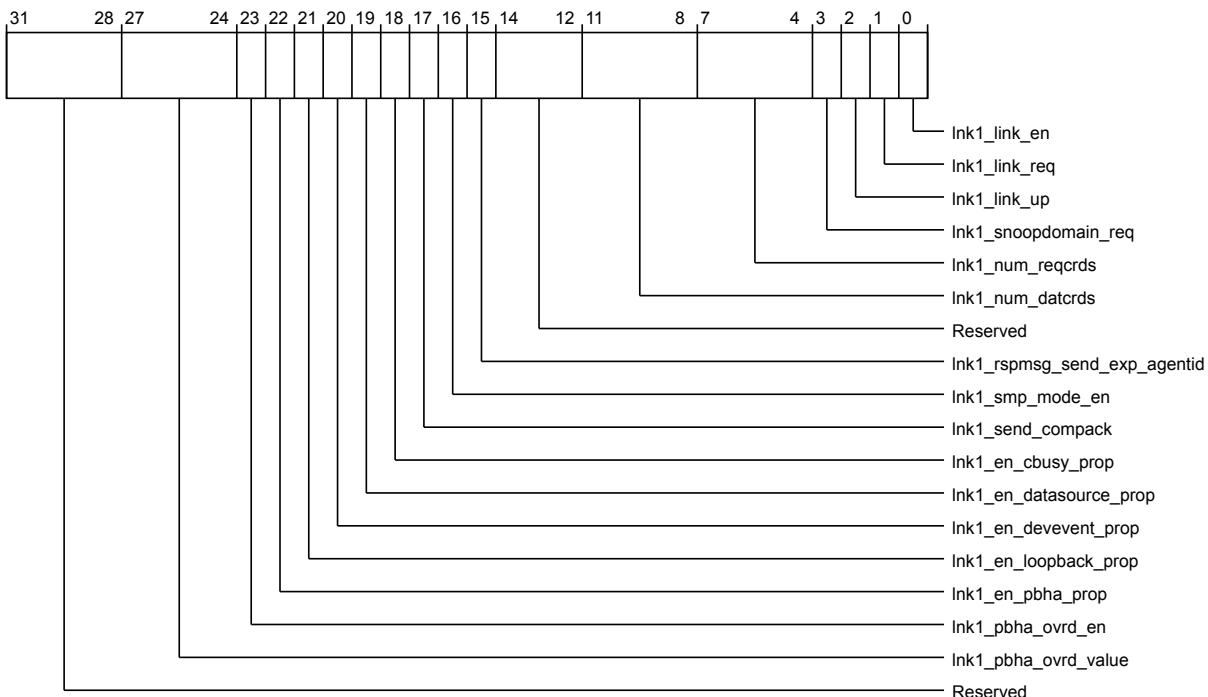


Figure 3-82 por_ccg_ha_por_ccg_ha_cxprtcl_link1_ctl (low)

The following table shows the por_ccg_ha_cxprtcl_link1_ctl lower register bit assignments.

Table 3-102 por_ccg_ha_por_ccg_ha_cxprtcl_link1_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	lnk1_pbha_ovrd_value	Override value for PBHA on CCIX Link 1. Applicable only when lnk1_pbha_ovrd_en is set. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	4'b0
23	lnk1_pbha_ovrd_en	When set, overrides PBHA on CCIX Link 1. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	1'b0
22	lnk1_en_pbha_prop	When set, enables propagation of PBHA on CCIX Link 1.	RW	1'b0
21	lnk1_en_loopback_prop	When set, enables propagation of LoopBack on CCIX Link 1.	RW	1'b0
20	lnk1_en_devevent_prop	When set, enables propagation of DevEvent on CCIX Link 1.	RW	1'b1
19	lnk1_en_datasource_prop	When set, enables propagation of DataSource on CCIX Link 1.	RW	1'b1
18	lnk1_en_cbusy_prop	When set, enables propagation of CBusy on CCIX Link 1.	RW	1'b1
17	lnk1_send_compack	When set, sends CompAck for CCIX Link 1.	RW	1'b0
16	lnk1_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 1.	RW	Configuration dependent
15	lnk1_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 1	RW	1'b0
14:12	Reserved	Reserved	RO	-
11:8	lnk1_num_datcrds	Controls the number of CCIX data credits assigned to Link 1 4'h0: Total credits equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
7:4	lnk1_num_reqrds	Controls the number of CCIX request credits assigned to Link 1 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0

Table 3-102 por_ccg_ha_por_ccg_ha_cxprtcl_link1_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
3	lnk1_snoopdomain_req	Controls Snoop domain enable (SYSCOREQ) for CCIX Link 1	RW	1'b0
2	lnk1_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk1_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk1_link_en	<p>Enables CCIX Link 1 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_ccg_ha_cxprtcl_link1_status

Functions as the CXHA CCIX Protocol Link 1 status register. Works with por_ccg_ha_cxprtcl_link1_ctl.
Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h1C18
Register reset	64'b010
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-83 por_ccg_ha_por_ccg_ha_cxprtcl_link1_status (high)

The following table shows the por_ccg_ha_cxprtcl_link1_status higher register bit assignments.

Table 3-103 por_ccg_ha_por_ccg_ha_cxprtcl_link1_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

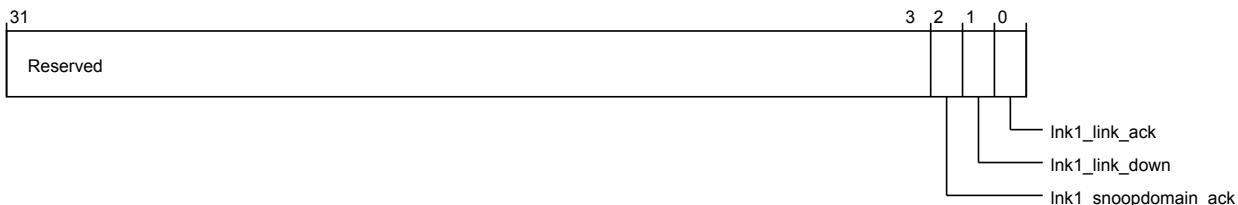


Figure 3-84 por_ccg_ha_por_ccg_ha_cxprtcl_link1_status (low)

The following table shows the por_ccg_ha_cxprtcl_link1_status lower register bit assignments.

Table 3-104 por_ccg_ha_por_ccg_ha_cxprtcl_link1_status (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	Lnk1_snoopdomain_ack	Provides Snoop domain status (SYSCOACK) for CCIX Link 1	RO	1'b0
1	Lnk1_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	Lnk1_link_ack	Link Up/Down Acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_ccg_ha_cxprtcl_link2_ctl

Functions as the CXHA CCIX Protocol Link 2 control register. Works with por_ccg_ha_cxprtcl_link2_status.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C20

Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

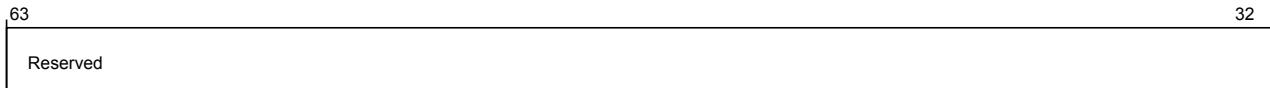


Figure 3-85 por_ccg_ha_por_ccg_ha_cxprtcl_link2_ctl (high)

The following table shows the por_ccg_ha_cxprtcl_link2_ctl higher register bit assignments.

Table 3-105 por_ccg_ha_por_ccg_ha_cxprtcl_link2_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

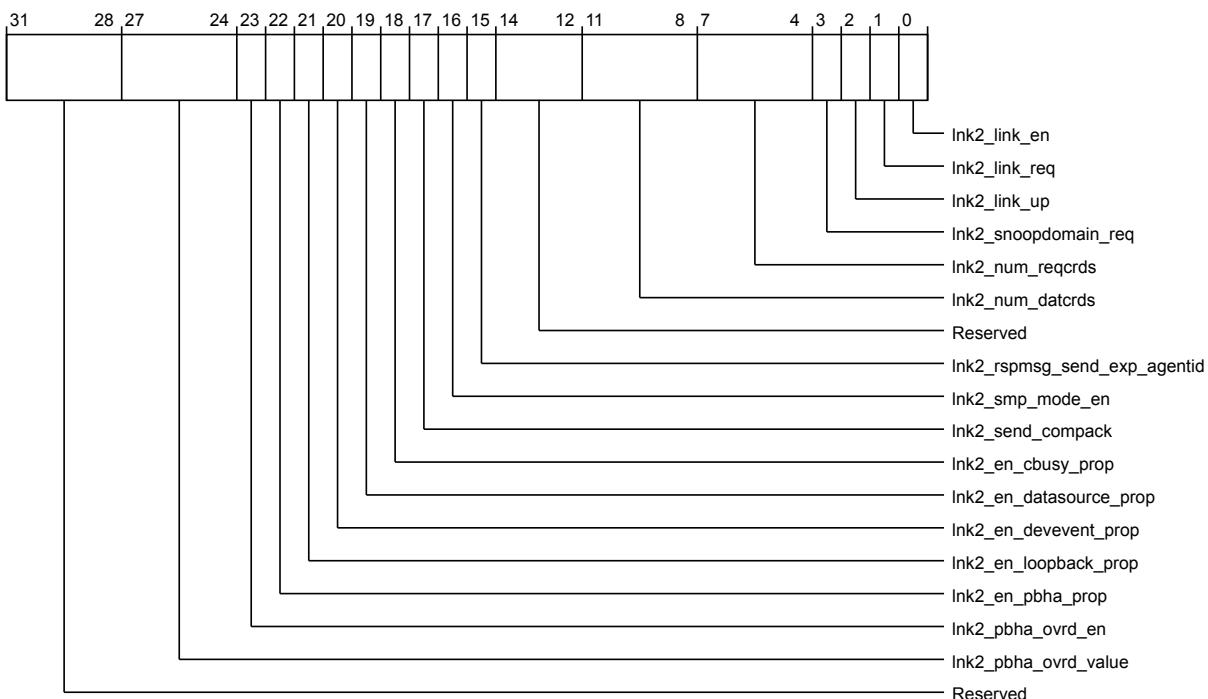


Figure 3-86 por_ccg_ha_por_ccg_ha_cxprtcl_link2_ctl (low)

The following table shows the por_ccg_ha_cxprtcl_link2_ctl lower register bit assignments.

Table 3-106 por_ccg_ha_por_ccg_ha_cxprtcl_link2_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	lnk2_pbha_ovrd_value	Override value for PBHA on CCIX Link 2. Applicable only when lnk2_pbha_ovrd_en is set. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	4'b0
23	lnk2_pbha_ovrd_en	When set, overrides PBHA on CCIX Link 2. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	1'b0
22	lnk2_en_pbha_prop	When set, enables propagation of PBHA on CCIX Link 2.	RW	1'b0
21	lnk2_en_loopback_prop	When set, enables propagation of LoopBack on CCIX Link 2.	RW	1'b0
20	lnk2_en_devevent_prop	When set, enables propagation of DevEvent on CCIX Link 2.	RW	1'b1
19	lnk2_en_datasource_prop	When set, enables propagation of DataSource on CCIX Link 2.	RW	1'b1
18	lnk2_en_cbusy_prop	When set, enables propagation of CBusy on CCIX Link 2.	RW	1'b1
17	lnk2_send_compack	When set, sends CompAck for CCIX Link 2.	RW	1'b0
16	lnk2_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 2.	RW	Configuration dependent
15	lnk2_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 2	RW	1'b0
14:12	Reserved	Reserved	RO	-
11:8	lnk2_num_datcrds	Controls the number of CCIX data credits assigned to Link 2 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
7:4	lnk2_num_reqrds	Controls the number of CCIX request credits assigned to Link 2 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0

Table 3-106 por_ccg_ha_por_ccg_ha_cxprtcl_link2_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
3	lnk2_snoopdomain_req	Controls Snoop domain enable (SYSCOREQ) for CCIX Link 2	RW	1'b0
2	lnk2_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk2_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk2_link_en	<p>Enables CCIX Link 2 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_ccg_ha_cxprtcl_link2_status

Functions as the CXHA CCIX Protocol Link 2 status register. Works with por_ccg_ha_cxprtcl_link2_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C28

Register reset 64'b010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

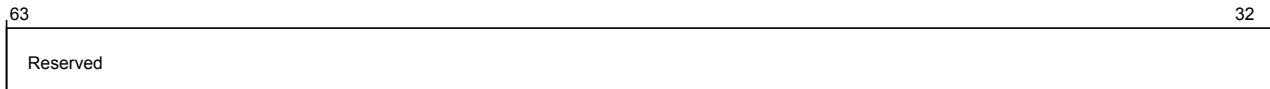


Figure 3-87 por_ccg_ha_por_ccg_ha_cxprtcl_link2_status (high)

The following table shows the por_ccg_ha_cxprtcl_link2_status higher register bit assignments.

Table 3-107 por_ccg_ha_por_ccg_ha_cxprtcl_link2_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

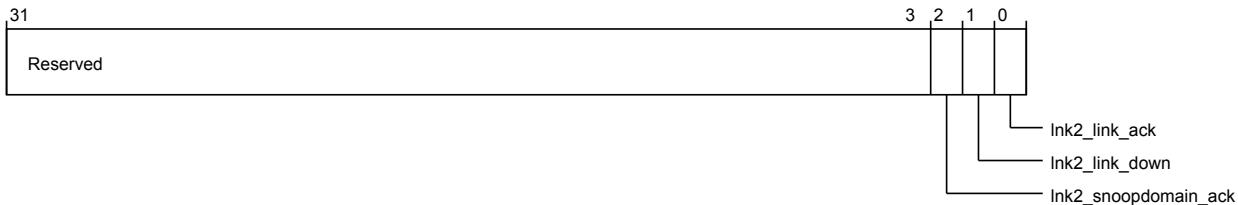


Figure 3-88 por_ccg_ha_por_ccg_ha_cxprtcl_link2_status (low)

The following table shows the por_ccg_ha_cxprtcl_link2_status lower register bit assignments.

Table 3-108 por_ccg_ha_por_ccg_ha_cxprtcl_link2_status (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	Lnk2_snoopdomain_ack	Provides Snoop domain status (SYSCOACK) for CCIX Link 2	RO	1'b0
1	Lnk2_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	Lnk2_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_ccg_ha_errfr

Functions as the error feature register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3000
Register reset	64'b00000010100101

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-89 por_ccg_ha_por_ccg_ha_errfr (high)

The following table shows the por_ccg_ha_errfr higher register bit assignments.

Table 3-109 por_ccg_ha_por_ccg_ha_errfr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

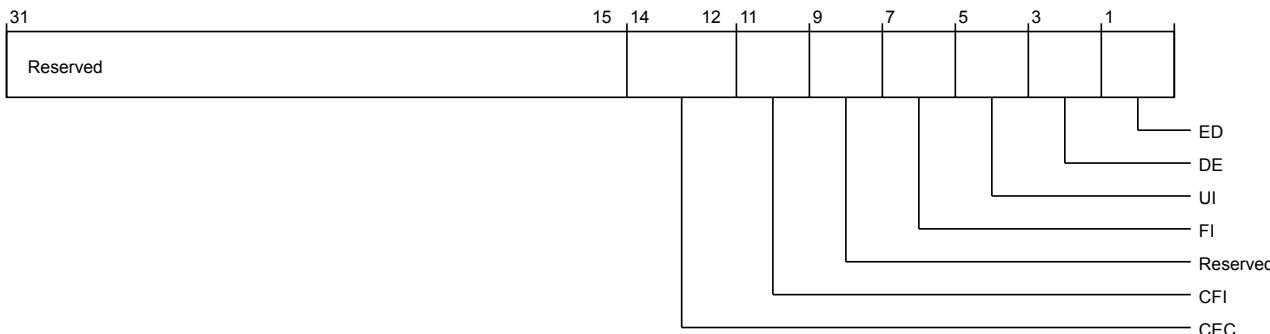


Figure 3-90 por_ccg_ha_por_ccg_ha_errfr (low)

The following table shows the por_ccg_ha_errfr lower register bit assignments.

Table 3-110 por_ccg_ha_por_ccg_ha_errfr (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_ccg_ha_errmisc[39:32] 3'b100: Implements 16-bit error counter in por_ccg_ha_errmisc[47:32]	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10

Table 3-110 por_ccg_ha_por_ccg_ha_errfr (low) (continued)

Bits	Field name	Description	Type	Reset
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_ccg_ha_errctlr

Functions as the error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3008
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group	por_ccg_ha_secure_register_groups_override.ras_secure_access_override_override

The following image shows the higher register bit assignments.



Figure 3-91 por_ccg_ha_por_ccg_ha_errctlr (high)

The following table shows the por_ccg_ha_errctlr higher register bit assignments.

Table 3-111 por_ccg_ha_por_ccg_ha_errctlr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

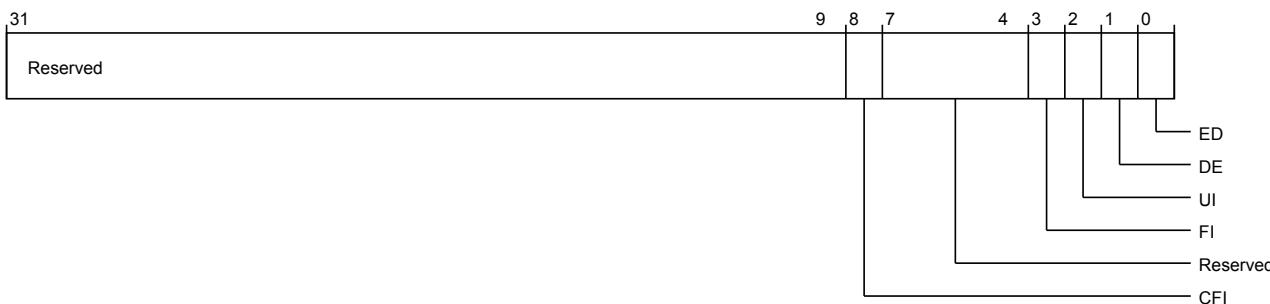


Figure 3-92 por_ccg_ha_por_ccg_ha_errctlr (low)

The following table shows the por_ccg_ha_errctlr lower register bit assignments.

Table 3-112 por_ccg_ha_por_ccg_ha_errctlr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_ccg_ha_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_ccg_ha_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_ccg_ha_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_ccg_ha_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_ccg_ha_errfr.ED	RW	1'b0

por_ccg_ha_errstatus

Functions as the error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3010
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-93 por_ccg_ha_por_ccg_ha_errstatus (high)

The following table shows the por_ccg_ha_errstatus higher register bit assignments.

Table 3-113 por_ccg_ha_por_ccg_ha_errstatus (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-94 por_ccg_ha_por_ccg_ha_errstatus (low)

The following table shows the por_ccg_ha_errstatus lower register bit assignments.

Table 3-114 por_ccg_ha_por_ccg_ha_errstatus (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_ccg_ha_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_ccg_ha_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-

Table 3-114 por_ccg_ha_por_ccg_ha_errstatus (low) (continued)

Bits	Field name	Description	Type	Reset
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_ccg_ha_erraddr

Contains the error record address.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3018

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

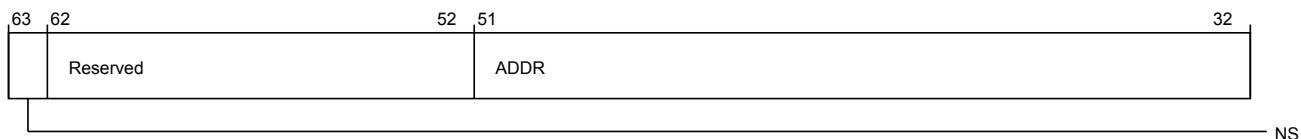


Figure 3-95 por_ccg_ha_por_ccg_ha_erraddr (high)

The following table shows the por_ccg_ha_erraddr higher register bit assignments.

Table 3-115 por_ccg_ha_por_ccg_ha_erraddr (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_ccg_ha_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.

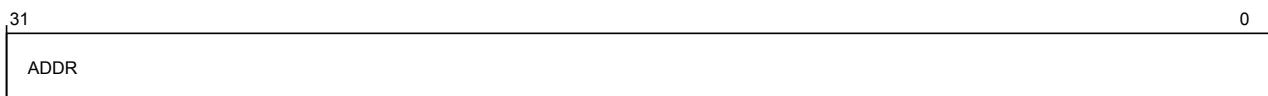


Figure 3-96 por_ccg_ha_por_ccg_ha_erraddr (low)

The following table shows the por_ccg_ha_erraddr lower register bit assignments.

Table 3-116 por_ccg_ha_por_ccg_ha_erraddr (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_ccg_ha_errmisc

Functions as the miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3020
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

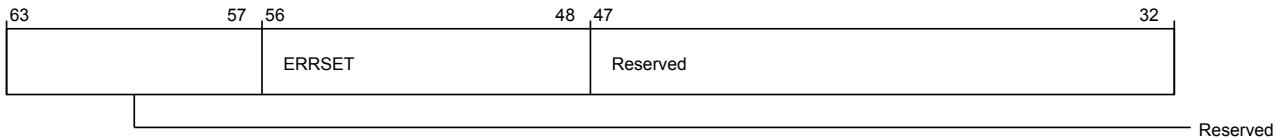


Figure 3-97 por_ccg_ha_por_ccg_ha_errmisc (high)

The following table shows the por_ccg_ha_errmisc higher register bit assignments.

Table 3-117 por_ccg_ha_por_ccg_ha_errmisc (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:48	ERRSET	RAM entry set address for parity error	RW	9'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

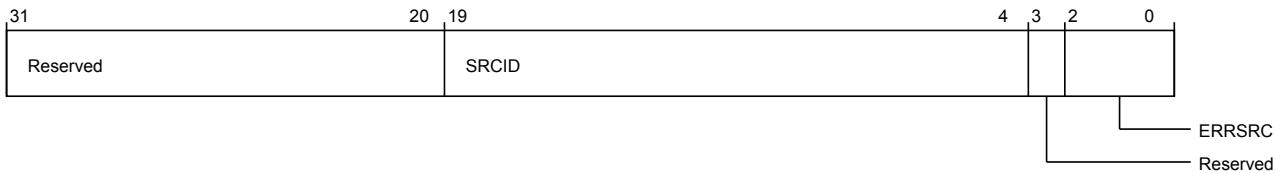


Figure 3-98 por_ccg_ha_por_ccg_ha_errmisc (low)

The following table shows the por_ccg_ha_errmisc lower register bit assignments.

Table 3-118 por_ccg_ha_por_ccg_ha_errmisc (low)

Bits	Field name	Description	Type	Reset
31:20	Reserved	Reserved	RO	-
19:4	SRCID	CCIX RAID of the requestor or the snoop target	RW	16'b0
3	Reserved	Reserved	RO	-
2:0	ERRSRC	Source of the parity error 3'b000: Read data buffer 0 3'b001: Read data buffer 1 3'b010: Write data buffer 0 3'b011: Write data buffer 1 3'b100: Passive Buffer	RW	3'b000

por_ccg_ha_errfr_NS

Functions as the non-secure error feature register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3100
Register reset	64'b0000010100101
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

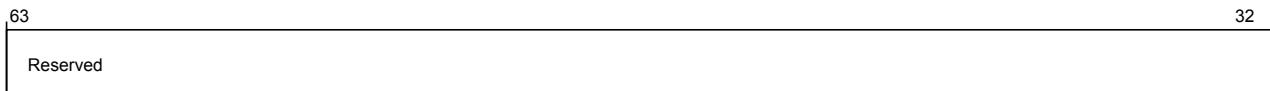


Figure 3-99 por_ccg_ha_por_ccg_ha_errfr_ns (high)

The following table shows the por_ccg_ha_errfr_NS higher register bit assignments.

Table 3-119 por_ccg_ha_por_ccg_ha_errfr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

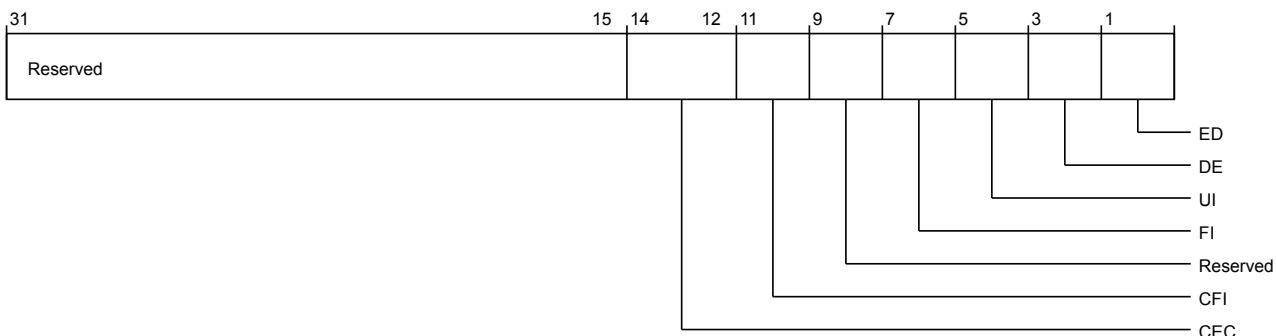


Figure 3-100 por_ccg_ha_por_ccg_ha_errfr_ns (low)

The following table shows the por_ccg_ha_errfr_NS lower register bit assignments.

Table 3-120 por_ccg_ha_por_ccg_ha_errfr_ns (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_ccg_ha_errmisc[39:32] 3'b100: Implements 16-bit error counter in por_ccg_ha_errmisc[47:32]	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_ccg_ha_errctlr_NS

Functions as the non-secure error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3108

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-101 por_ccg_ha_por_ccg_ha_errctlr_ns (high)

The following table shows the por_ccg_ha_errctlr_NS higher register bit assignments.

Table 3-121 por_ccg_ha_por_ccg_ha_errctlr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

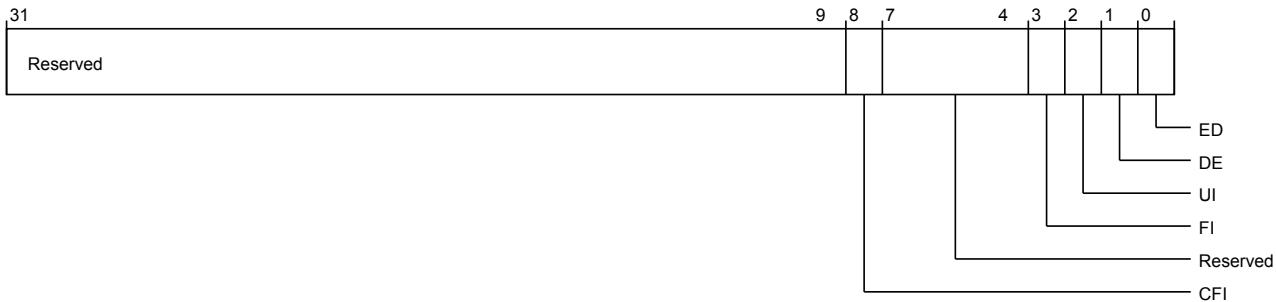


Figure 3-102 por_ccg_ha_por_ccg_ha_errctr_ns (low)

The following table shows the por_ccg_ha_errctlr_NS lower register bit assignments.

Table 3-122 por_ccg_ha_por_ccg_ha_errctr_ns (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_ccg_ha_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_ccg_ha_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_ccg_ha_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_ccg_ha_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_ccg_ha_errfr.ED	RW	1'b0

por_ccg_ha_errstatus_NS

Functions as the non-secure error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h3

Register reset 64'b0

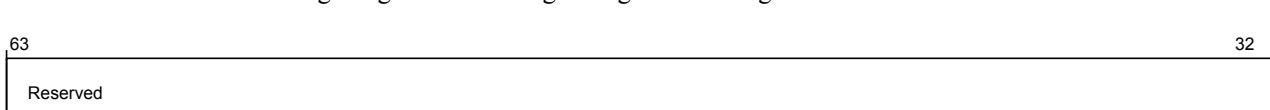


Figure 3-103 por ccq ha por ccq ha errstatus ns (high)

The following table shows the por_ccg_ha_errstatus_NS higher register bit assignments.

Table 3-123 por_ccg_ha_por_ccg_ha_errstatus_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-104 por_ccg_ha_por_ccg_ha_errstatus_ns (low)

The following table shows the por_ccg_ha_errstatus_NS lower register bit assignments.

Table 3-124 por_ccg_ha_por_ccg_ha_errstatus_ns (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_ccg_ha_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0

Table 3-124 por_ccg_ha_por_ccg_ha_errstatus_ns (low) (continued)

Bits	Field name	Description	Type	Reset
26	MV	por_ccg_ha_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_ccg_ha_erraddr_NS

Contains the non-secure error record address.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3118

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

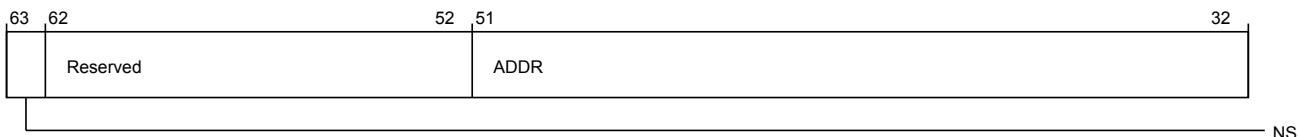


Figure 3-105 por_ccg_ha_por_ccg_ha_erraddr_ns (high)

The following table shows the por_ccg_ha_erraddr_NS higher register bit assignments.

Table 3-125 por_ccg_ha_por_ccg_ha_erraddr_ns (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_ccg_ha_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.

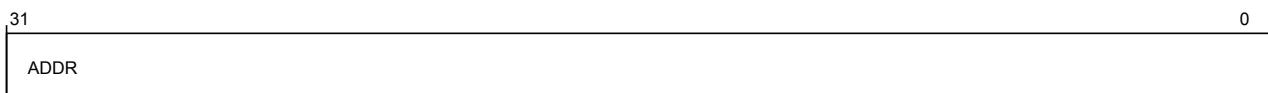


Figure 3-106 por_ccg_ha_por_ccg_ha_erraddr_ns (low)

The following table shows the por_ccg_ha_erraddr_NS lower register bit assignments.

Table 3-126 por_ccg_ha_por_ccg_ha_erraddr_ns (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_ccg_ha_errmisc_NS

Functions as the non-secure miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3120

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

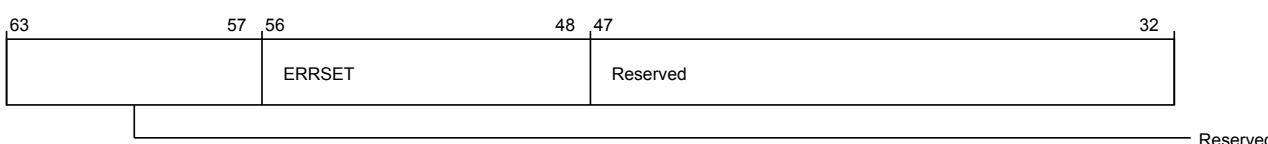


Figure 3-107 por_ccg_ha_por_ccg_ha_errmisc_ns (high)

The following table shows the por_ccg_ha_errmisc_NS higher register bit assignments.

Table 3-127 por_ccg_ha_por_ccg_ha_errmisc_ns (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:48	ERRSET	RAM entry set address for parity error	RW	9'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

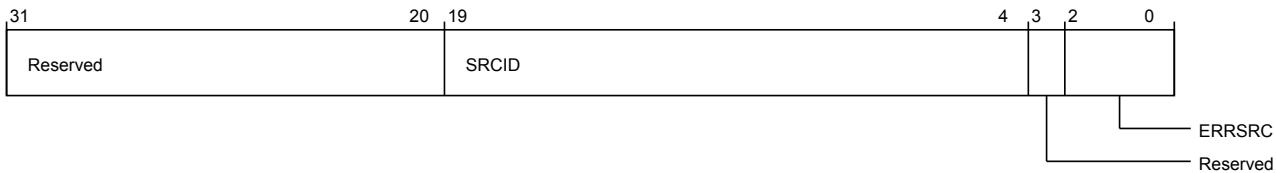


Figure 3-108 por_ccg_ha_por_ccg_ha_errmisc_ns (low)

The following table shows the por_ccg_ha_errmisc_NS lower register bit assignments.

Table 3-128 por_ccg_ha_por_ccg_ha_errmisc_ns (low)

Bits	Field name	Description	Type	Reset
31:20	Reserved	Reserved	RO	-
19:4	SRCID	CCIX RAID of the requestor or the snoop target	RW	16'b0
3	Reserved	Reserved	RO	-
2:0	ERRSRC	Source of the parity error 3'b000: Read data buffer 0 3'b001: Read data buffer 1 3'b010: Write data buffer 0 3'b011: Write data buffer 1 3'b100: Passive Buffer	RW	3'b000

3.3.3 CXRA register descriptions

This section lists the CXRA registers.

por_cxg_ra_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset	16'h0
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

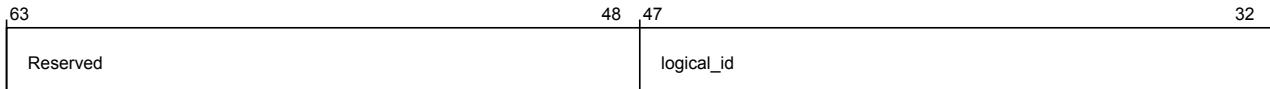


Figure 3-109 por_cxg_ra_por_cxg_ra_node_info (high)

The following table shows the por_cxg_ra_node_info higher register bit assignments.

Table 3-129 por_cxg_ra_por_cxg_ra_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

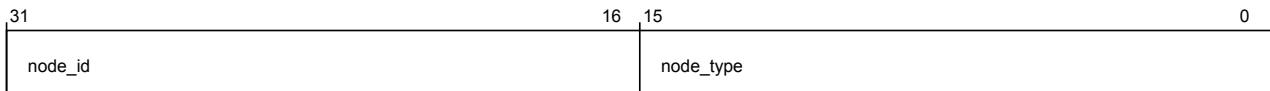


Figure 3-110 por_cxg_ra_por_cxg_ra_node_info (low)

The following table shows the por_cxg_ra_node_info lower register bit assignments.

Table 3-130 por_cxg_ra_por_cxg_ra_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0100

por_cxg_ra_child_info

Provides component child identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h80
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

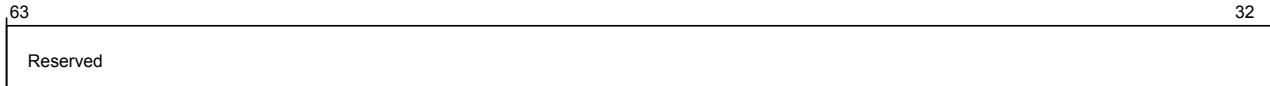


Figure 3-111 por_cxg_ra_por_cxg_ra_child_info (high)

The following table shows the por_cxg_ra_child_info higher register bit assignments.

Table 3-131 por_cxg_ra_por_cxg_ra_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

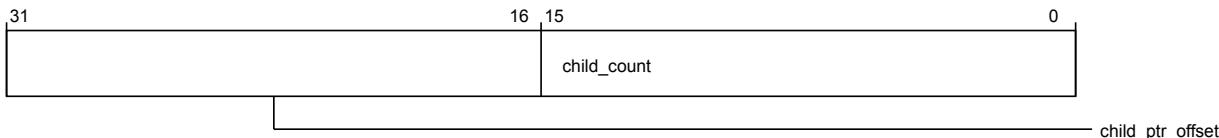


Figure 3-112 por_cxg_ra_por_cxg_ra_child_info (low)

The following table shows the por_cxg_ra_child_info lower register bit assignments.

Table 3-132 por_cxg_ra_por_cxg_ra_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'h0

por_cxg_ra_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

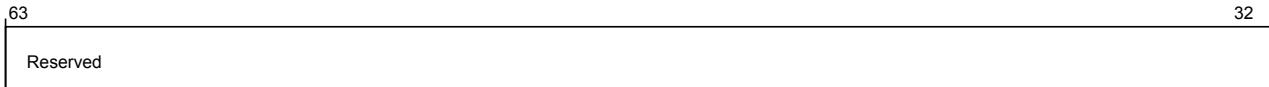


Figure 3-113 por_cxg_ra_por_cxg_ra_secure_register_groups_override (high)

The following table shows the por_cxg_ra_secure_register_groups_override higher register bit assignments.

Table 3-133 por_cxg_ra_por_cxg_ra_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

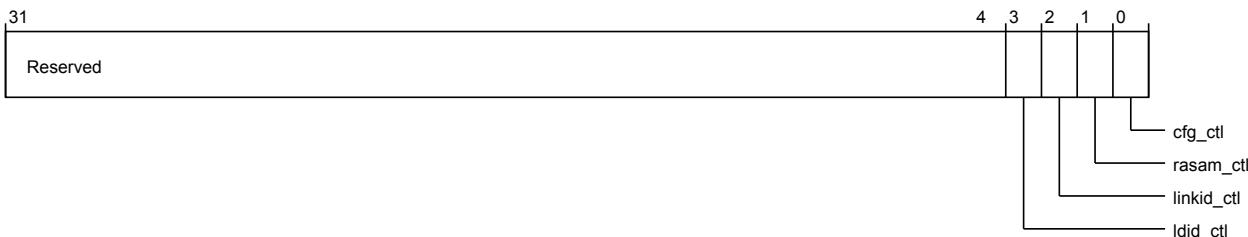


Figure 3-114 por_cxg_ra_por_cxg_ra_secure_register_groups_override (low)

The following table shows the por_cxg_ra_secure_register_groups_override lower register bit assignments.

Table 3-134 por_cxg_ra_por_cxg_ra_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	ldid_ctl	Allows non-secure access to secure RA LDID registers	RW	1'b0
2	linkid_ctl	Allows non-secure access to secure RA Link ID registers	RW	1'b0
1	rasam_ctl	Allows non-secure access to secure RA SAM control registers	RW	1'b0
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_cxg_ra_unit_info

Provides component identification information for CXRA.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

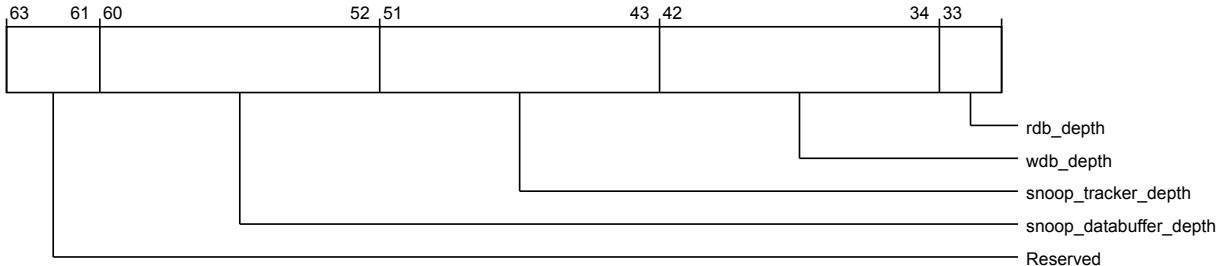


Figure 3-115 por_cxg_ra_por_cxg_ra_unit_info (high)

The following table shows the por_cxg_ra_unit_info higher register bit assignments.

Table 3-135 por_cxg_ra_por_cxg_ra_unit_info (high)

Bits	Field name	Description	Type	Reset
63:61	Reserved	Reserved	RO	-
60:52	snoop_databuffer_depth	Depth of Snoop Data Buffer - number of outstanding SNP requests on CHI	RO	Configuration dependent
51:43	snoop_tracker_depth	Depth of Snoop Tracker - number of outstanding SNP requests on CCIX	RO	Configuration dependent
42:34	wdb_depth	Depth of Write Data Buffer	RO	Configuration dependent
33:32	rdb_depth	Depth of Read Data Buffer	RO	Configuration dependent

The following image shows the lower register bit assignments.

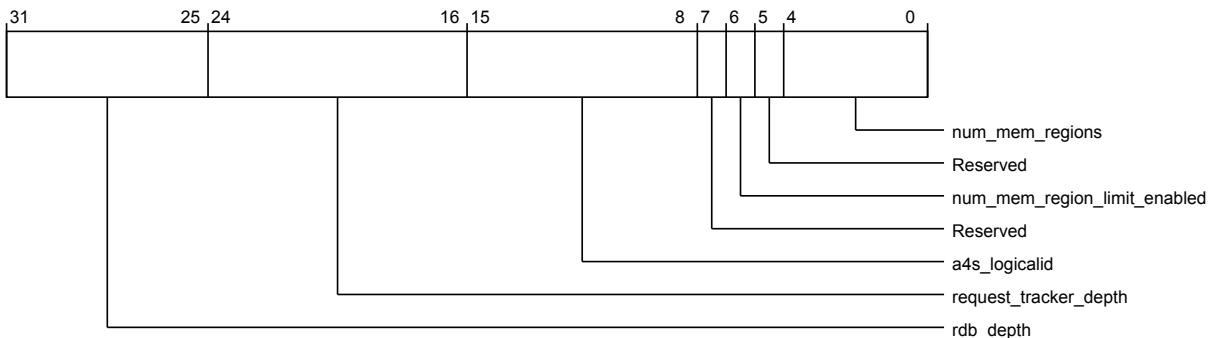


Figure 3-116 por_cxg_ra_por_cxg_ra_unit_info (low)

The following table shows the por_cxg_ra_unit_info lower register bit assignments.

Table 3-136 por_cxg_ra_por_cxg_ra_unit_info (low)

Bits	Field name	Description	Type	Reset
31:25	rdb_depth	Depth of Read Data Buffer	RO	Configuration dependent
24:16	request_tracker_depth	Depth of Request Tracker - number of outstanding Memory requests on CCIX	RO	Configuration dependent
15:8	a4s_logicalid	AXI4Stream interfaces logical ID	RO	Configuration dependent
7	Reserved	Reserved	RO	-
6	num_mem_region_limit_enabled	Memory region limiting enabled	RO	Configuration dependent
5	Reserved	Reserved	RO	-
4:0	num_mem_regions	Number of memory regions supported	RO	Configuration dependent

por_cxg_ra_cfg_ctl

Functions as the configuration control register. Specifies the current mode.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA00

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_cxg_ra_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

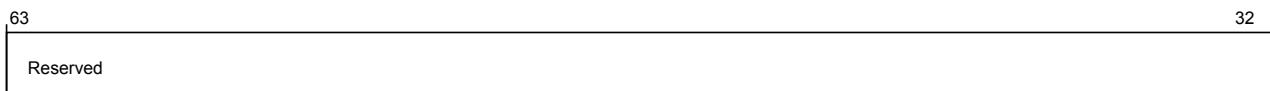


Figure 3-117 por_cxg_ra_por_cxg_ra_cfg_ctl (high)

The following table shows the por_cxg_ra_cfg_ctl higher register bit assignments.

Table 3-137 por_cxg_ra_por_cxg_ra_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

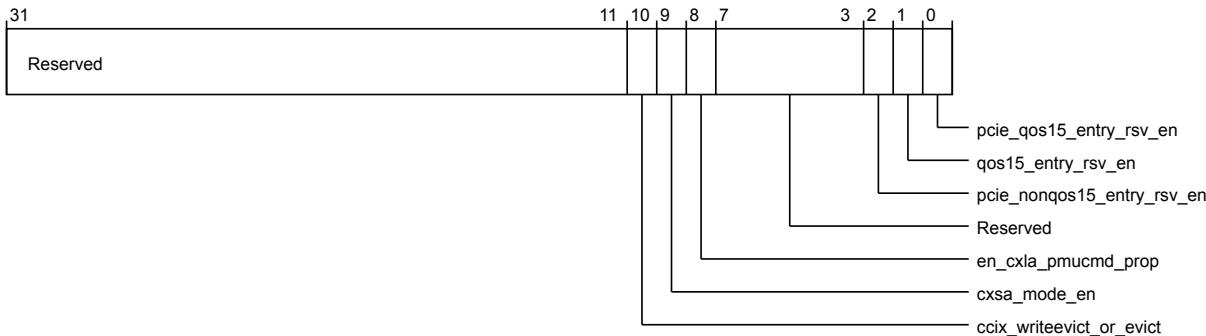


Figure 3-118 por_cxg_ra_por_cxg_ra_cfg_ctl (low)

The following table shows the por_cxg_ra_cfg_ctl lower register bit assignments.

Table 3-138 por_cxg_ra_por_cxg_ra_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10	ccix_writeevict_or_evict	When set, downgrades WriteEvict to Evict 1'b1: Evict is sent instead of WriteEvict 1'b0: WriteEvict is sent	RW	1'b0
9	cxsa_mode_en	When set, enables the CCIX Slave Agent mode. In this mode RA functions as a CCIX Slave Agent 1'b1: CCIX Slave Agent 1'b0: CCIX Requesting Agent	RW	1'b0
8	en_cxla_pmucmd_prop	When set, enables the propagation of PMU commands to CXLA NOTE: By default, CXLA PMU command propagation is disabled.	RW	1'b0
7:3	Reserved	Reserved	RO	-
2	pcie_nonqos15_entry_rsv_en	Enables entry reservation for non QoS15 traffic from PCIe RN-I/RN-D 1'b1: Reserves tracker entry for non QoS15 requests from PCIe RN-I/RN-D 1'b0: Does not reserve tracker entry for non QoS15 requests from PCIe RN-I/RN-D	RW	Configuration dependent

Table 3-138 por_cxg_ra_por_cxg_ra_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
1	qos15_entry_rsv_en	Enables entry reservation for QoS15 traffic 1'b1: Reserves tracker entry for QoS15 requests 1'b0: Does not reserve tracker entry for QoS15 requests	RW	1'b1
0	pcie_qos15_entry_rsv_en	Enables entry reservation for QoS15 traffic from PCIe RN-I/RN-D 1'b1: Reserves tracker entry for QoS15 requests from PCIe RN-I/RN-D 1'b0: Does not reserve tracker entry for QoS15 requests from PCIe RN-I/RN-D	RW	Configuration dependent

por_cxg_ra_aux_ctl

Functions as the auxiliary control register for CXRA.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	64'b110000000000110
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-119 por_cxg_ra_por_cxg_ra_aux_ctl (high)

The following table shows the por_cxg_ra_aux_ctl higher register bit assignments.

Table 3-139 por_cxg_ra_por_cxg_ra_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

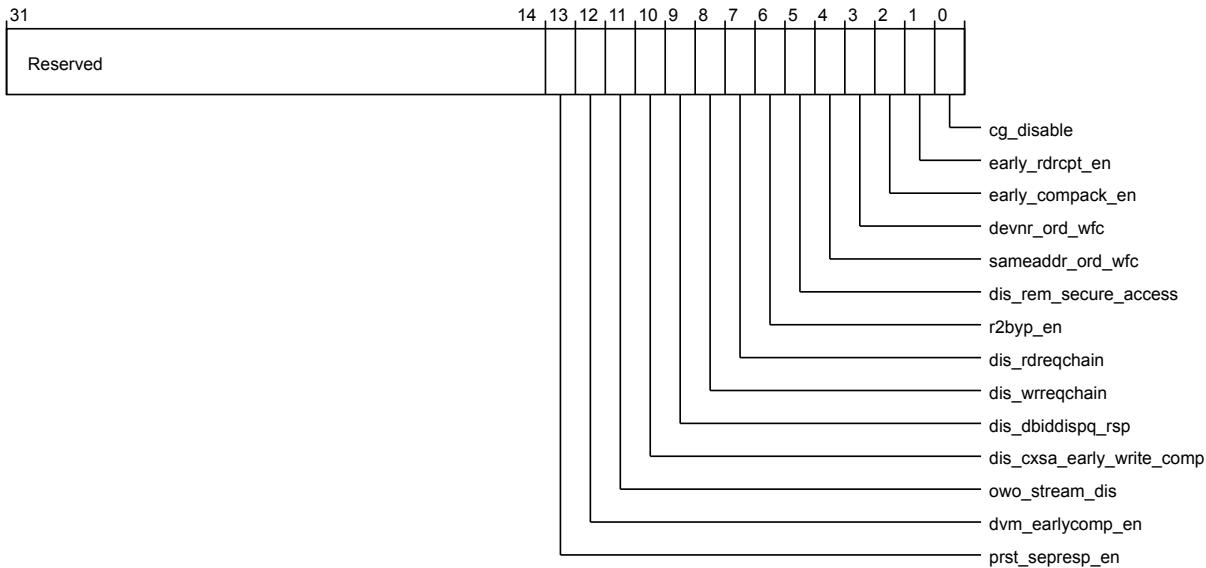


Figure 3-120 por_cxg_ra_por_cxg_ra_aux_ctl (low)

The following table shows the por_cxg_ra_aux_ctl lower register bit assignments.

Table 3-140 por_cxg_ra_por_cxg_ra_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:14	Reserved	Reserved	RO	-
13	prst_sepresp_en	When set, enables separate persist response on CCIX for persistent cache maintenance (PCMO2) operation. Note: this bit is applicable only in SMP mode.	RW	1'b1
12	dvm_earlycomp_en	When set, enables early DVM Op completion responses from RA.	RW	1'b1
11	owo_stream_dis	When set, disables CompAck dependency to dispatch an ordered PCIe write.	RW	1'b0
10	dis_cxs_a_early_write_comp	When set, disables early write completions in CCIX Slave Agent mode.	RW	1'b0
9	dis_dbiddispq_rsp	When set, disables the dispatch of DBID responses from a separate DispatchQ.	RW	1'b0
8	dis_wrreqchain	When set, disables chaining of write requests.	RW	1'b0
7	dis_rdreqchain	When set, disables chaining of read and dataless requests.	RW	1'b0
6	r2byp_en	When set, enables request bypass. Applies to read and dataless requests only. Note: When set will affect the capability to chain a request on the TX side	RW	1'b0
5	dis_rem_secure_access	When set, treats all the incoming snoops as non-secure and forces the NS bit to 1	RW	1'b0
4	sameaddr_ord_wfc	When set, enables waiting for completion (COMP) before dispatching next same Addr dependent transaction (TXN)	RW	1'b0
3	devnr_ord_wfc	When set, enables waiting for completion (COMP) before dispatching next Device-nR dependent transaction (TXN)	RW	1'b0

Table 3-140 por_cxg_ra_por_cxg_ra_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	early_compack_en	Early CompAck enable; enables sending early CompAck on CCIX for requests that require CompAck	RW	1'b1
1	early_rdrcpt_en	Early ReadReceipt enable; enables sending early ReadReceipt for ordered read requests	RW	1'b1
0	cg_disable	Disables clock gating when set	RW	1'b0

por_cxg_ra_cbusy_limit_ctl

Cbusy threshold limits for RHT entries.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA18
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

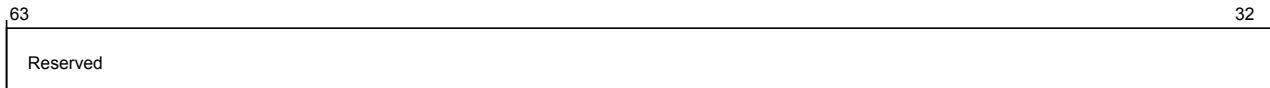


Figure 3-121 por_cxg_ra_por_cxg_ra_cbusy_limit_ctl (high)

The following table shows the por_cxg_ra_cbusy_limit_ctl higher register bit assignments.

Table 3-141 por_cxg_ra_por_cxg_ra_cbusy_limit_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

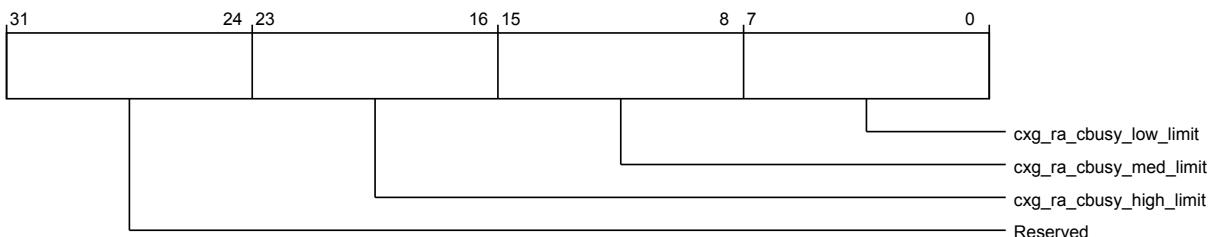


Figure 3-122 por_cxg_ra_por_cxg_ra_cbusy_limit_ctl (low)

The following table shows the por_cxg_ra_cbusy_limit_ctl lower register bit assignments.

Table 3-142 por_cxg_ra_por_cxg_ra_cbusy_limit_ctl (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	cxg_ra_cbusy_high_limit	RHT limit for CBusy High	RW	Configuration dependent
15:8	cxg_ra_cbusy_med_limit	RHT limit for CBusy Med	RW	Configuration dependent
7:0	cxg_ra_cbusy_low_limit	RHT limit for CBusy Low	RW	Configuration dependent

por_cxg_ra_sam_addr_region_reg_0-7

This register repeats 7 times. It parametrized by the i from 0 to 7. Configures Address Region #*{i}* for RA SAM.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + # <i>{8*[0, 1, 2, 3, 4, 5, 6, 7]}</i>
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ra_secure_register_groups_override.rasam_ctl

The following image shows the higher register bit assignments.

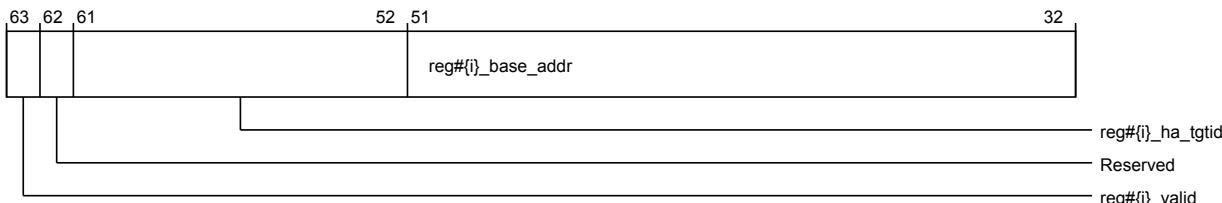


Figure 3-123 por_cxg_ra_por_cxg_ra_sam_addr_region_reg_0-7 (high)

The following table shows the por_cxg_ra_sam_addr_region_reg_0-7 higher register bit assignments.

Table 3-143 por_cxg_ra_por_cxg_ra_sam_addr_region_reg_0-7 (high)

Bits	Field name	Description	Type	Reset
63	reg#{i}_valid	Specifies if the memory region is valid	RW	1'b0
62	Reserved	Reserved	RO	-
61:52	reg#{i}_ha_tgtid	Specifies the target HAID	RW	10'b0
51:32	reg#{i}_base_addr	Specifies the 2^n-aligned base address for the memory region	RW	36'h0

The following image shows the lower register bit assignments.

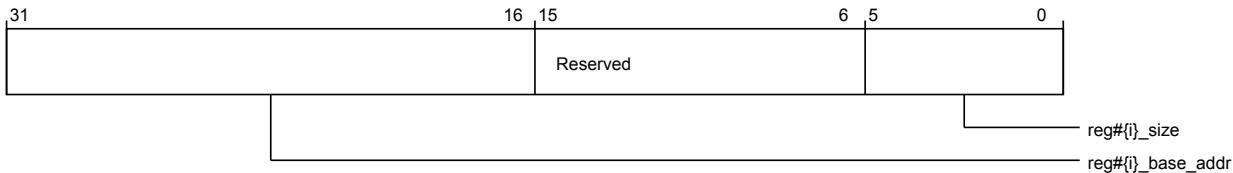


Figure 3-124 por_cxg_ra_por_cxg_ra_sam_addr_region_reg_0-7 (low)

The following table shows the por_cxg_ra_sam_addr_region_reg_0-7 lower register bit assignments.

Table 3-144 por_cxg_ra_por_cxg_ra_sam_addr_region_reg_0-7 (low)

Bits	Field name	Description	Type	Reset
31:16	reg#{i}_base_addr	Specifies the 2^n -aligned base address for the memory region	RW	36'h0
15:6	Reserved	Reserved	RO	-
5:0	reg#{i}_size	Specifies the size of the memory region	RW	1'b0

por_cxg_ra_agentid_to_linkid_val

Specifies which Agent ID to Link ID mappings are valid.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ra_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

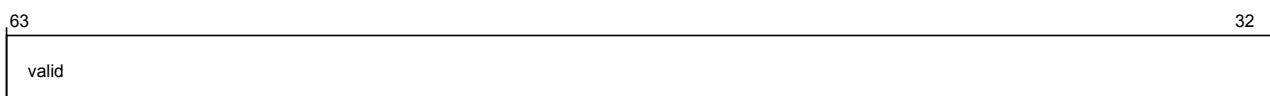


Figure 3-125 por_cxg_ra_por_cxg_ra_agentid_to_linkid_val (high)

The following table shows the por_cxg_ra_agentid_to_linkid_val higher register bit assignments.

Table 3-145 por_cxg_ra_por_cxg_ra_agentid_to_linkid_val (high)

Bits	Field name	Description	Type	Reset
63:32	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

The following image shows the lower register bit assignments.

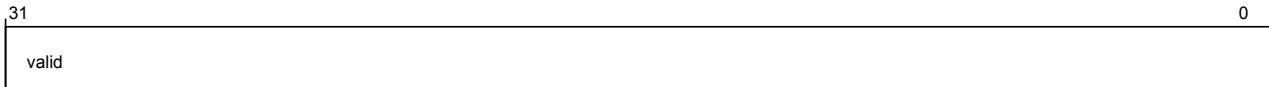


Figure 3-126 por_cxg_ra_por_cxg_ra_agentid_to_linkid_val (low)

The following table shows the por_cxg_ra_agentid_to_linkid_val lower register bit assignments.

Table 3-146 por_cxg_ra_por_cxg_ra_agentid_to_linkid_val (low)

Bits	Field name	Description	Type	Reset
31:0	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

por_cxg_ra_agentid_to_linkid_reg_0-7

This register repeats 7 times. It parametrized by the i from 0 to 7. Specifies the mapping of Agent ID to Link ID for Agent IDs # $\{i*8\}$ to # $\{i*8+7\}$.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD10 + # $\{[0, 1, 2, 3, 4, 5, 6, 7]*8\}$

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ra_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

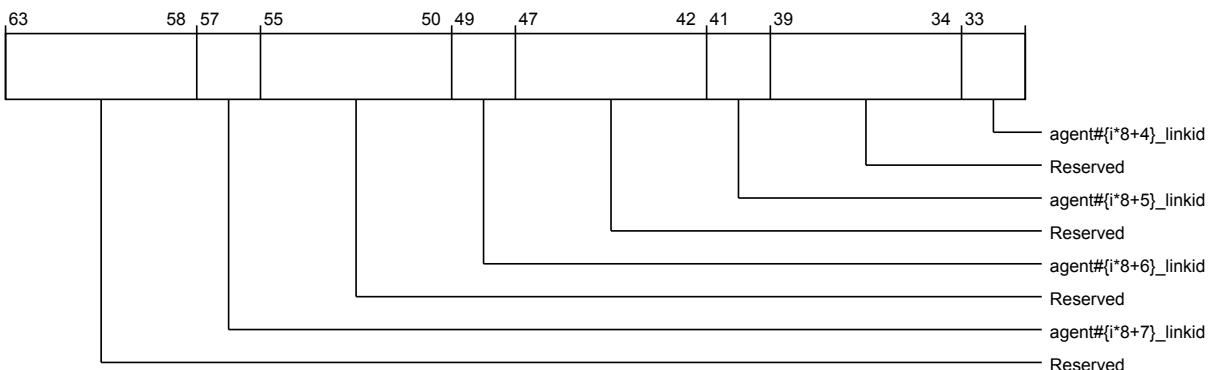


Figure 3-127 por_cxg_ra_por_cxg_ra_agentid_to_linkid_reg_0-7 (high)

The following table shows the por_cxg_ra_agentid_to_linkid_reg_0-7 higher register bit assignments.

Table 3-147 por_cxg_ra_por_cxg_ra_agentid_to_linkid_reg_0-7 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent#{i*8+7}_linkid	Specifies the Link ID for Agent ID # $\{i*8+7\}$	RW	2'h0

Table 3-147 por_cxg_ra_por_cxg_ra_agentid_to_linkid_reg_0-7 (high) (continued)

Bits	Field name	Description	Type	Reset
55:50	Reserved	Reserved	RO	-
49:48	agent#{i*8+6}_linkid	Specifies the Link ID for Agent ID #{i*8+6}	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent#{i*8+5}_linkid	Specifies the Link ID for Agent ID #{i*8+5}	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent#{i*8+4}_linkid	Specifies the Link ID for Agent ID #{i*8+4}	RW	2'h0

The following image shows the lower register bit assignments.

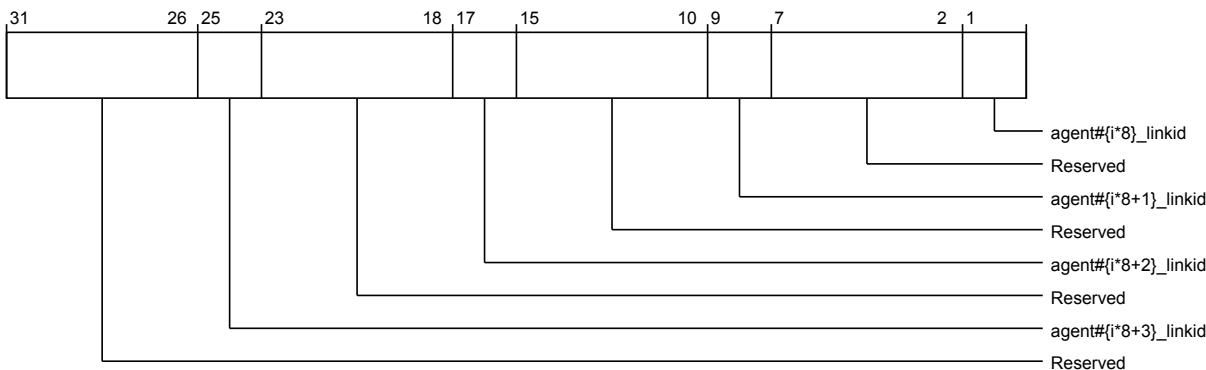


Figure 3-128 por_cxg_ra_por_cxg_ra_agentid_to_linkid_reg_0-7 (low)

The following table shows the por_cxg_ra_agentid_to_linkid_reg_0-7 lower register bit assignments.

Table 3-148 por_cxg_ra_por_cxg_ra_agentid_to_linkid_reg_0-7 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent#{i*8+3}_linkid	Specifies the Link ID for Agent ID #{i*8+3}	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent#{i*8+2}_linkid	Specifies the Link ID for Agent ID #{i*8+2}	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent#{i*8+1}_linkid	Specifies the Link ID for Agent ID #{i*8+1}	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent#{i*8}_linkid	Specifies the Link ID for Agent ID #{i*8}	RW	2'h0

por_cxg_ra_rni_lidid_to_exp_raid_reg_0-9

This register repeats 9 times. It parametrized by the i from 0 to 9. Specifies the mapping of RN-Is LDID to Expanded RAID for LDIDs # $\{1^*4\}$ to # $\{i^*4+3\}$.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE00 + # {[0, 1, 2, .., 8, 9]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

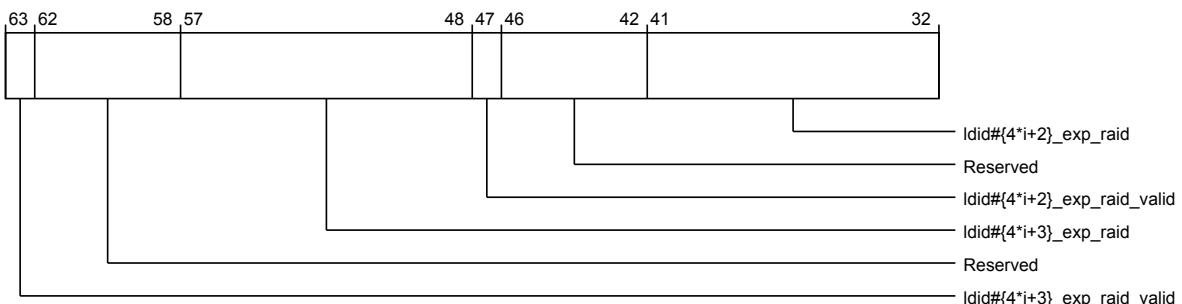


Figure 3-129 por_cxg_ra_por_cxg_ra_rni_lidid_to_exp_raid_reg_0-9 (high)

The following table shows the port expansion register bit assignments.

Table 3-149 por_cxg_ra_por_cxg_ra_rni_lidid_to_exp_raid_reg_0-9 (high)

Bits	Field name	Description	Type	Reset
63	ldid#{4*i+3}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i+3} is valid;	RW	1'h0
62:58	Reserved	Reserved	RO	-
57:48	ldid#{4*i+3}_exp_raid	Specifies the Expanded RAID for LDID #{4*i+3}	RW	10'h0
47	ldid#{4*i+2}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i+2} is valid;	RW	1'h0
46:42	Reserved	Reserved	RO	-
41:32	ldid#{4*i+2}_exp_raid	Specifies the Expanded RAID for LDID #{4*i+2}	RW	10'h0

The following image shows the lower register bit assignments.

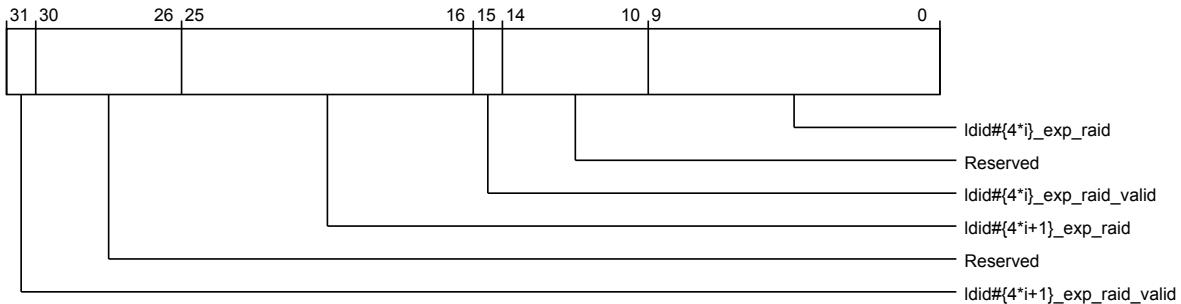


Figure 3-130 por_cxg_ra_por_cxg_ra_rni_ldid_to_exp_raid_reg_0-9 (low)

The following table shows the por_cxg_ra_rni_ldid_to_exp_raid_reg_0-9 lower register bit assignments.

Table 3-150 por_cxg_ra_por_cxg_ra_rni_ldid_to_exp_raid_reg_0-9 (low)

Bits	Field name	Description	Type	Reset
31	ldid#{4*i+1}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i+1} is valid;	RW	1'h0
30:26	Reserved	Reserved	RO	-
25:16	ldid#{4*i+1}_exp_raid	Specifies the Expanded RAID for LDID #{4*i+1}	RW	10'h0
15	ldid#{4*i}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i} is valid;	RW	1'h0
14:10	Reserved	Reserved	RO	-
9:0	ldid#{4*i}_exp_raid	Specifies the Expanded RAID for LDID #{4*i}	RW	10'h0

por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9

This register repeats 9 times. It parametrized by the i from 0 to 9. Specifies the mapping of RN-D's LDID to Expanded RAID for LDIDs #{{i}*4} to #{{i}*4+3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF00 + #{{[0, 1, 2, ..., 8, 9]}*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

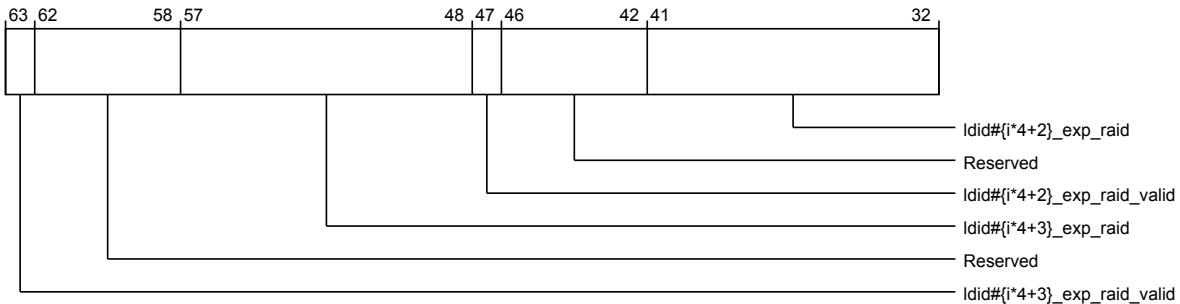


Figure 3-131 por_cxg_ra_por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9 (high)

The following table shows the por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9 higher register bit assignments.

Table 3-151 por_cxg_ra_por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9 (high)

Bits	Field name	Description	Type	Reset
63	ldid#{i*4+3}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4+3} is valid;	RW	1'h0
62:58	Reserved	Reserved	RO	-
57:48	ldid#{i*4+3}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+3}	RW	10'h0
47	ldid#{i*4+2}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4+2} is valid;	RW	1'h0
46:42	Reserved	Reserved	RO	-
41:32	ldid#{i*4+2}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+2}	RW	10'h0

The following image shows the lower register bit assignments.

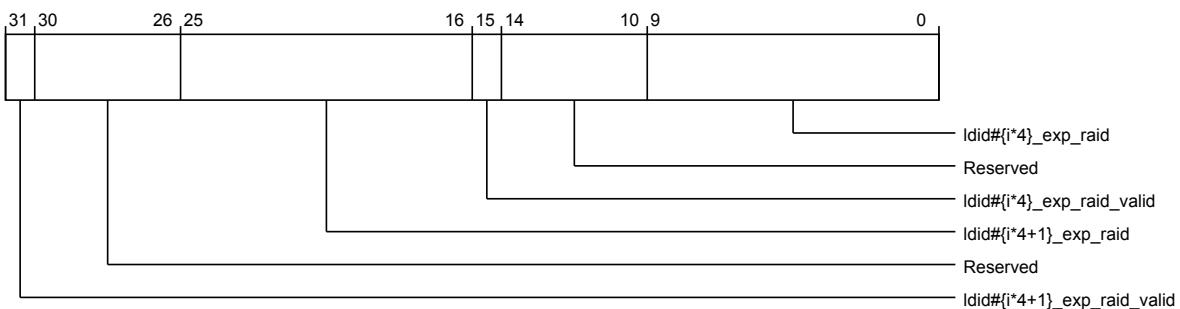


Figure 3-132 por_cxg_ra_por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9 (low)

The following table shows the por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9 lower register bit assignments.

Table 3-152 por_cxg_ra_por_cxg_ra_rnd_ldid_to_exp_raid_reg_0-9 (low)

Bits	Field name	Description	Type	Reset
31	ldid#{i*4+1}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4+1} is valid;	RW	1'h0
30:26	Reserved	Reserved	RO	-
25:16	ldid#{i*4+1}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+1}	RW	10'h0

Table 3-152 por_cxg_ra_por_cxg_ra_rnd_Idid_to_exp_raid_reg_0-9 (low) (continued)

Bits	Field name	Description	Type	Reset
15	ldid#{i*4}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4} is valid;	RW	1'h0
14:10	Reserved	Reserved	RO	-
9:0	ldid#{i*4}_exp_raid	Specifies the Expanded RAID for LDID #{i*4}	RW	10'h0

por_cxg_ra_rnf_lid_to_exp_raid_reg_0-127

This register repeats 127 times. It parametrized by the i from 0 to 127. Specifies the mapping of RN-F's LDID to Expanded RAID for LDIDs # $\{i^*4\}$ to # $\{i^*4+3\}$.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset $16'h1000 + \# \{ [0, 1, 2, \dots, 126, 127] * 8 \}$

Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

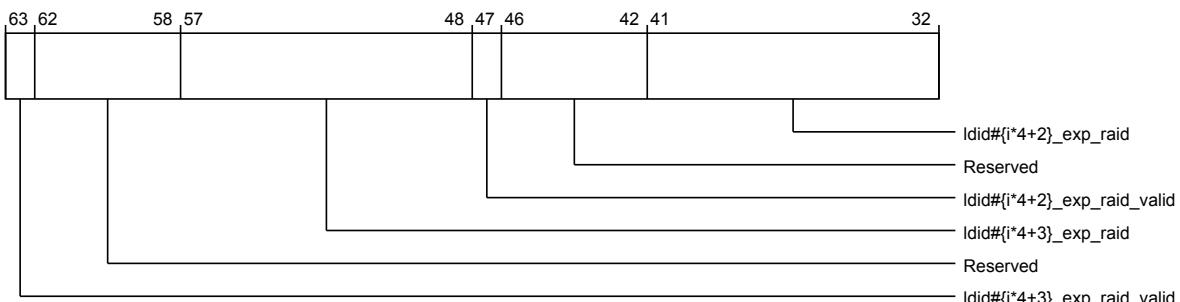


Figure 3-133 por cxq ra por cxq ra rnf ldid to exp raid req 0-127 (high)

The following table shows the por_cxg_ra_rnf_ldid_to_exp_raid_reg_0-127 higher register bit assignments.

Table 3-153 por cxq ra por cxq ra rnf ldid to exp raid req 0-127 (high)

Bits	Field name	Description	Type	Reset
63	ldid#{i*4+3}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4+3} is valid;	RW	1'h0
62:58	Reserved	Reserved	RO	-
57:48	ldid#{i*4+3}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+3}	RW	10'h0
47	ldid#{i*4+2}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4+2} is valid;	RW	1'h0

Table 3-153 por_cxg_ra_por_cxg_ra_rnf_ldid_to_exp_raid_reg_0-127 (high) (continued)

Bits	Field name	Description	Type	Reset
46:42	Reserved	Reserved	RO	-
41:32	ldid#{i*4+2}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+2}	RW	10'h0

The following image shows the lower register bit assignments.

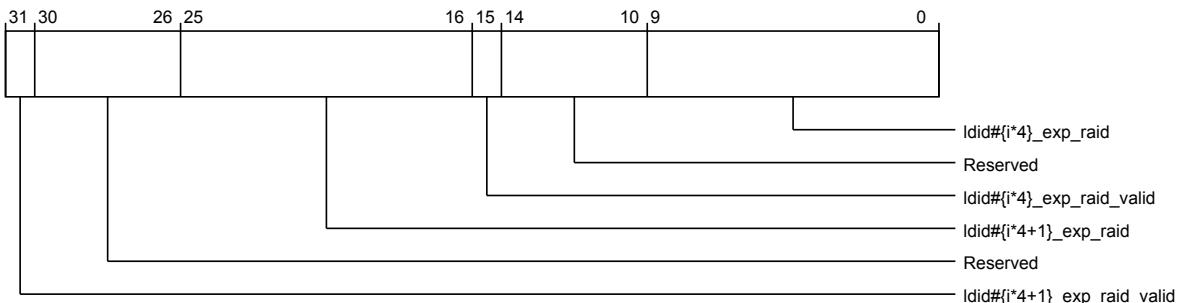


Figure 3-134 por_cxg_ra_por_cxg_ra_rnf_ldid_to_exp_raid_reg_0-127 (low)

The following table shows the por_cxg_ra_rnf_ldid_to_exp_raid_reg_0-127 lower register bit assignments.

Table 3-154 por_cxg_ra_por_cxg_ra_rnf_ldid_to_exp_raid_reg_0-127 (low)

Bits	Field name	Description	Type	Reset
31	ldid#{i*4+1}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4+1} is valid;	RW	1'h0
30:26	Reserved	Reserved	RO	-
25:16	ldid#{i*4+1}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+1}	RW	10'h0
15	ldid#{i*4}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4} is valid;	RW	1'h0
14:10	Reserved	Reserved	RO	-
9:0	ldid#{i*4}_exp_raid	Specifies the Expanded RAID for LDID #{i*4}	RW	10'h0

por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127

This register repeats 127 times. It parametrized by the i from 0 to 127. Specifies the mapping of RN-F's Default LDID to CHI NodeID for LDIDs #{i*4} to #{i*4+3}.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h1400 + #[0, 1, 2, .., 126, 127]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

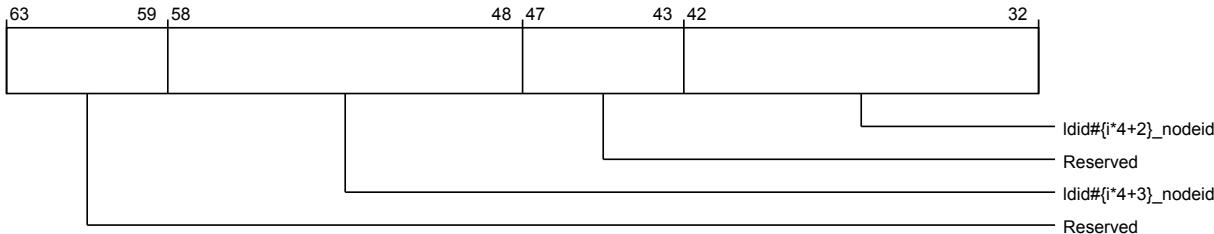


Figure 3-135 por_cxg_ra_por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127 (high)

The following table shows the por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127 higher register bit assignments.

Table 3-155 por_cxg_ra_por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	ldid#{i*4+3}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4+3\}$	RO	11'h0
47:43	Reserved	Reserved	RO	-
42:32	ldid#{i*4+2}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4+2\}$	RO	11'h0

The following image shows the lower register bit assignments.

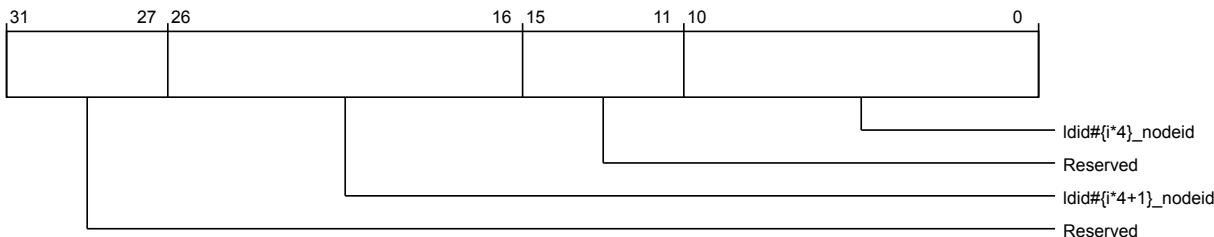


Figure 3-136 por_cxg_ra_por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127 (low)

The following table shows the por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127 lower register bit assignments.

Table 3-156 por_cxg_ra_por_cxg_ra_rnf_ldid_to_nodeid_reg_0-127 (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:16	ldid#{i*4+1}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4+1\}$	RO	11'h0
15:11	Reserved	Reserved	RO	-
10:0	ldid#{i*4}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4\}$	RO	11'h0

por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127

This register repeats 127 times. It parametrized by the i from 0 to 127. Specifies the mapping of RN-F's overridden LDID for default LDIDs # $\{i*4\}$ to # $\{i*4+3\}$. Valid only if POR_MXP_RNF_CLUSTER_EN_PARAM is 1

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1800 + # $\{[0, 1, 2, \dots, 126, 127]*8\}$
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

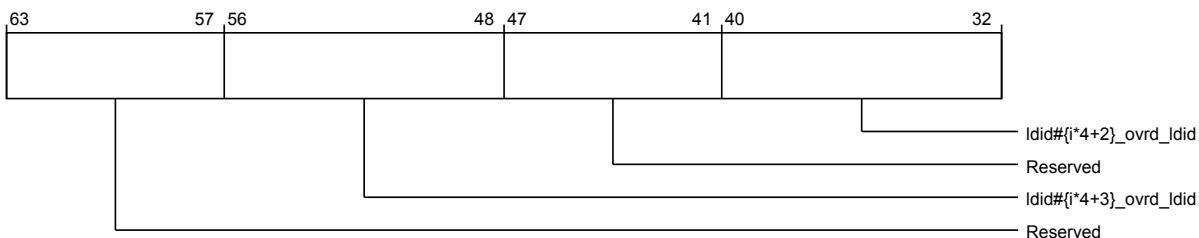


Figure 3-137 por_cxg_ra_por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (high)

The following table shows the por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 higher register bit assignments.

Table 3-157 por_cxg_ra_por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:48	ldid#{i*4+3}_ovrd_ldid	Specifies the Overridden LDID for Default LDID # $\{i*4+3\}$	RW	Configuration dependent
47:41	Reserved	Reserved	RO	-
40:32	ldid#{i*4+2}_ovrd_ldid	Specifies the Overridden LDID for Default LDID # $\{i*4+2\}$	RW	Configuration dependent

The following image shows the lower register bit assignments.

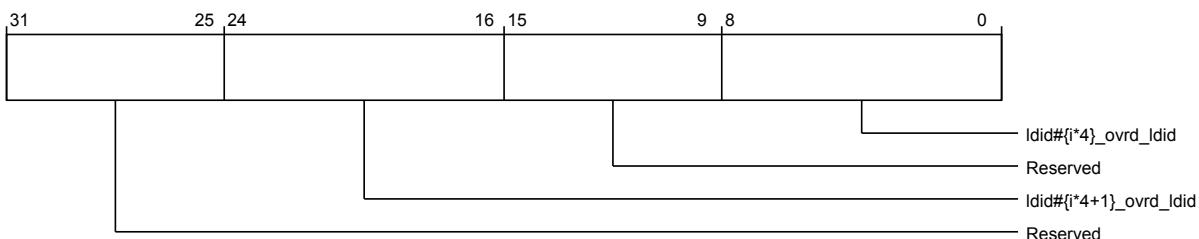


Figure 3-138 por_cxg_ra_por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (low)

The following table shows the por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 lower register bit assignments.

Table 3-158 por_cxg_ra_por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24:16	ldid#{i*4+1}_ovrd_ldid	Specifies the Overridden LDID for Default LDID #{i*4+1}	RW	Configuration dependent
15:9	Reserved	Reserved	RO	-
8:0	ldid#{i*4}_ovrd_ldid	Specifies the Overridden LDID for Default LDID #{i*4}	RW	Configuration dependent

por_cxg_ra_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

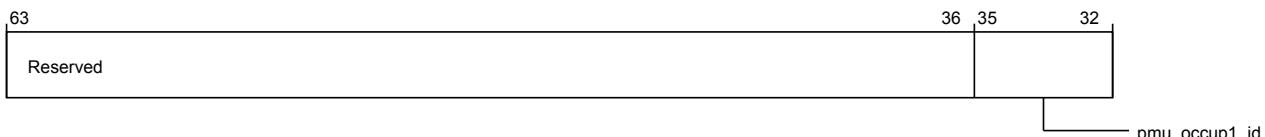


Figure 3-139 por_cxg_ra_por_cxg_ra_pmu_event_sel (high)

The following table shows the por_cxg_ra_pmu_event_sel higher register bit assignments.

Table 3-159 por_cxg_ra_por_cxg_ra_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:36	Reserved	Reserved	RO	-
35:32	pmu_occup1_id	PMU occupancy event selector ID	RW	4'b0

The following image shows the lower register bit assignments.

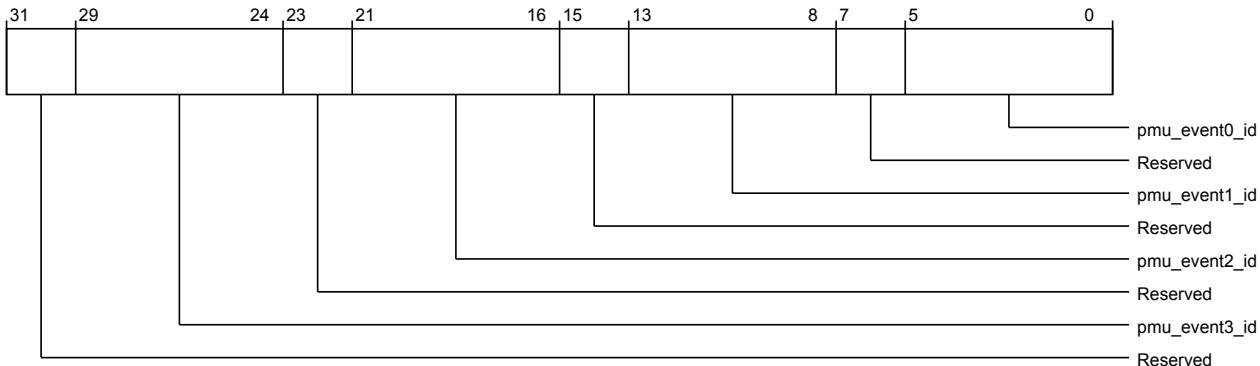


Figure 3-140 por_cxg_ra_por_cxg_ra_pmu_event_sel (low)

The following table shows the por_cxg_ra_pmu_event_sel lower register bit assignments.

Table 3-160 por_cxg_ra_por_cxg_ra_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	CXRA PMU Event 3 ID; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	CXRA PMU Event 2 ID; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	CXRA PMU Event 1 ID; see pmu_event0_id for encodings	RW	6'b0

Table 3-160 por_cxg_ra_por_cxg_ra_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-
5:0	pmu_event0_id	CXRA PMU Event 0 ID 6'h00: No event 6'h01: Request Tracker (RHT) occupancy count overflow 6'h02: Snoop Tracker (SHT) occupancy count overflow 6'h03: Read Data Buffer (RDB) occupancy count overflow 6'h04: Write Data Buffer (WDB) occupancy count overflow 6'h05: Snoop Sink Buffer (SSB) occupancy count overflow 6'h06: CCIX RX broadcast snoops 6'h07: CCIX TX request chain 6'h08: CCIX TX request chain average length 6'h09: CHI internal RSP stall 6'h0A: CHI internal DAT stall 6'h0B: CCIX REQ Protocol credit Link 0 stall 6'h0C: CCIX REQ Protocol credit Link 1 stall 6'h0D: CCIX REQ Protocol credit Link 2 stall 6'h0E: CCIX DAT Protocol credit Link 0 stall 6'h0F: CCIX DAT Protocol credit Link 1 stall 6'h10: CCIX DAT Protocol credit Link 2 stall 6'h11: CHI external RSP stall 6'h12: CHI external DAT stall 6'h13: CCIX MISC Protocol credit Link 0 stall 6'h14: CCIX MISC Protocol credit Link 1 stall 6'h15: CCIX MISC Protocol credit Link 2 stall	RW	6'b0

por_cxg_ra_cxprtcl_link0_ctl

Functions as the CXRA CCIX Protocol Link 0 control register. Works with por_cxg_ra_cxprtcl_link0_status.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C00
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63
Reserved

32

Figure 3-141 por_cxg_ra_por_cxg_ra_cxpctl_link0_ctl (high)

The following table shows the por_cxg_ra_cxpctl_link0_ctl higher register bit assignments.

Table 3-161 por_cxg_ra_por_cxg_ra_cxpctl_link0_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

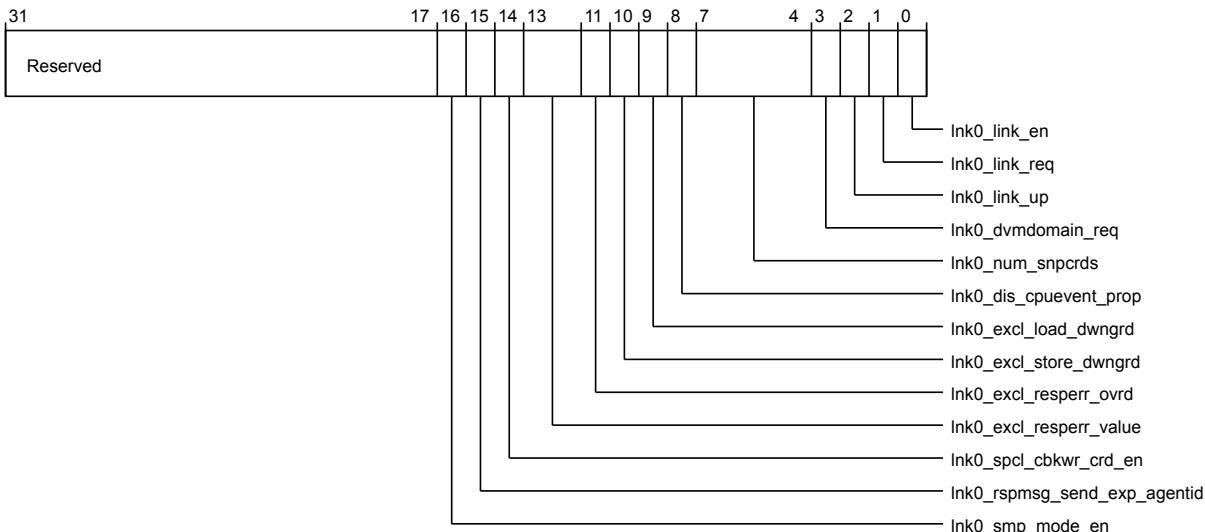


Figure 3-142 por_cxg_ra_por_cxg_ra_cxpctl_link0_ctl (low)

The following table shows the por_cxg_ra_cxpctl_link0_ctl lower register bit assignments.

Table 3-162 por_cxg_ra_por_cxg_ra_cxpctl_link0_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	lnk0_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 0.	RW	Configuration dependent
15	lnk0_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 0	RW	1'b0
14	lnk0_spcl_cbkwr_crd_en	When set, notifies RA to use special credits from HA to send CopyBack writes on CCIX Link 0 NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0

Table 3-162 por_cxg_ra_por_cxg_ra_cxpctl_link0_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
13:12	lnk0_excl_resperr_value	<p>Two bit value to override RespErr field of an exclusive response. Applicable only if lnk0_excl_resperr_ovrd bit is set.</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	2'b0
11	lnk0_excl_resperr_ovrd	<p>When set, overrides the RespErr field of exclusive response with the lnk0_excl_resperr_value field</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
10	lnk0_excl_store_dwngrd	<p>When set, downgrades shareable exclusive store to shareable store when sending on CCIX Link 0</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
9	lnk0_excl_load_dwngrd	<p>When set, downgrades shareable exclusive load to shareable load when sending on CCIX Link 0</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
8	lnk0_dis_cpuevent_prop	<p>When set, disables the propagation of CPU Events on CCIX Link 0</p> <p>NOTE: This field is applicable only when SMP Mode enable is set.</p>	RW	1'b0
7:4	lnk0_num_snpercds	<p>Controls the number of CCIX snoop credits assigned to Link 0</p> <p>4'h0: Total credits are equally divided across all links</p> <p>4'h1: 25% of credits assigned</p> <p>4'h2: 50% of credits assigned</p> <p>4'h3: 75% of credits assigned</p> <p>4'h4: 100% of credits assigned</p> <p>4'hF: 0% of credits assigned</p>	RW	4'b0
3	lnk0_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for CCIX Link 0	RW	1'b0

Table 3-162 por_cxg_ra_por_cxg_ra_cxpctl_link0_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	lnk0_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk0_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk0_link_en	<p>Enables CCIX Link 0 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_cxg_ra_cxpctl_link0_status

Functions as the CXRA CCIX Protocol Link 0 status register. Works with por_cxg_ra_cxpctl_link0_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C08

Register reset 64'b0010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-143 por_cxg_ra_por_cxg_ra_cxpctl_link0_status (high)

The following table shows the por_cxg_ra_cxpctl_link0_status higher register bit assignments.

Table 3-163 por_cxg_ra_por_cxg_ra_cxprtcl_link0_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

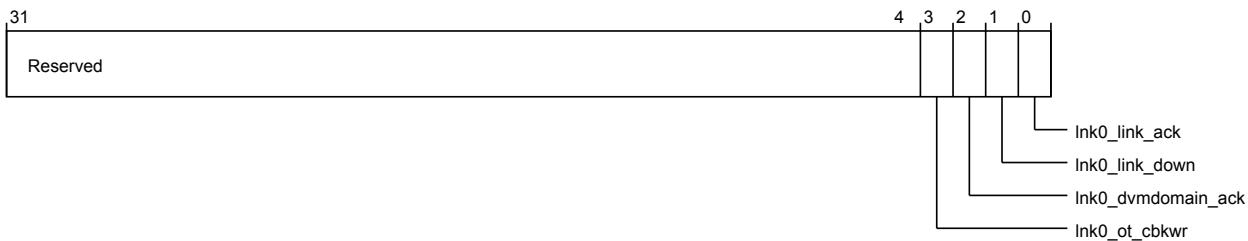


Figure 3-144 por_cxg_ra_por_cxg_ra_cxprtcl_link0_status (low)

The following table shows the por_cxg_ra_cxprtcl_link0_status lower register bit assignments.

Table 3-164 por_cxg_ra_por_cxg_ra_cxprtcl_link0_status (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	lnk0_ot_cbkwr	Provides status for outstanding CopyBack Write for CCIX Link0	RO	1'b0
2	lnk0_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for CCIX Link 0	RO	1'b0
1	lnk0_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	lnk0_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops granting protocol credits and starts returning protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_cxg_ra_cxprtcl_link1_ctl

Functions as the CXRA CCIX Protocol Link 1 control register. Works with por_cxg_ra_cxprtcl_link1_status.

Its characteristics are:

Type	RW
Register width (Bits)	64

Address offset	16'h1C10
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

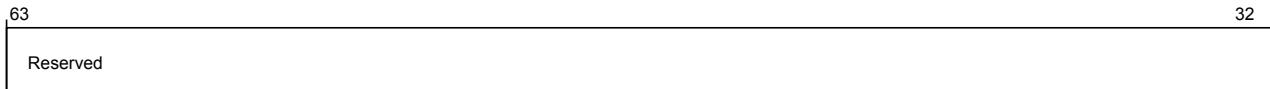


Figure 3-145 por_cxg_ra_por_cxg_ra_cxpctl_link1_ctl (high)

The following table shows the por_cxg_ra_cxpctl_link1_ctl higher register bit assignments.

Table 3-165 por_cxg_ra_por_cxg_ra_cxpctl_link1_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

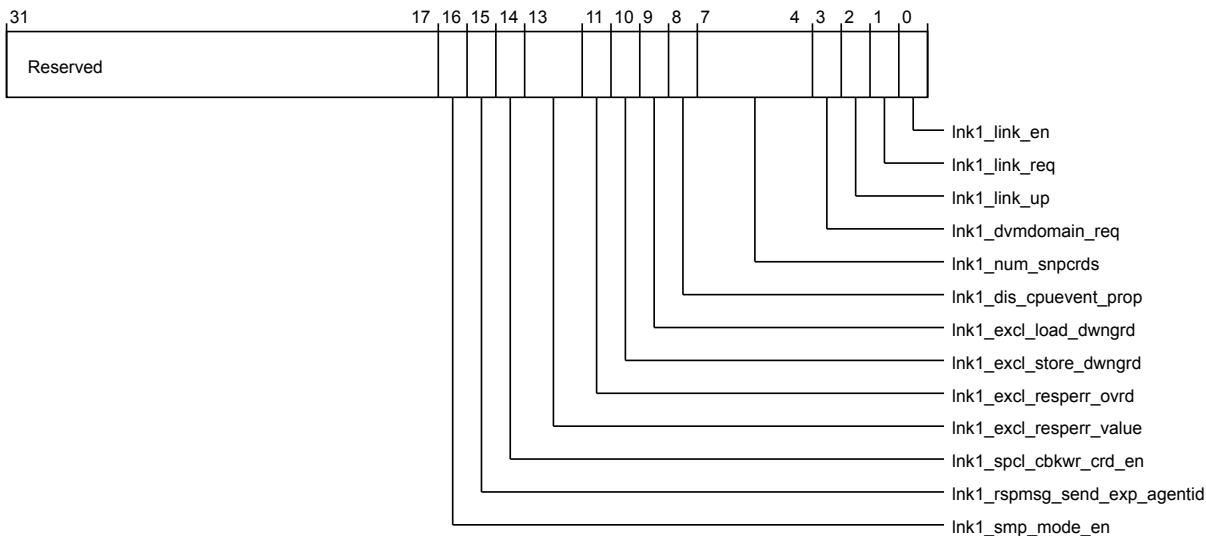


Figure 3-146 por_cxg_ra_por_cxg_ra_cxpctl_link1_ctl (low)

The following table shows the por_cxg_ra_cxpctl_link1_ctl lower register bit assignments.

Table 3-166 por_cxg_ra_por_cxg_ra_cxpctl_link1_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	lnk1_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 1.	RW	Configuration dependent

Table 3-166 por_cxg_ra_por_cxg_ra_cxpctl_link1_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
15	lnk1_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 1	RW	1'b0
14	lnk1_spcl_cbkwr_crd_en	When set, notifies RA to use special credits from HA to send CopyBack writes on CCIX Link 1 NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
13:12	lnk1_excl_resperr_value	Two bit value to override RespErr field of an exclusive response. Applicable only if lnk1_excl_resperr_ovrd bit is set. NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	2'b0
11	lnk1_excl_resperr_ovrd	When set, overrides the RespErr field of exclusive response with the lnk1_excl_resperr_value field NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	1'b0
10	lnk1_excl_store_dwngrd	When set, downgrades shareable exclusive store to shareable store when sending on CCIX Link 1 NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	1'b0
9	lnk1_excl_load_dwngrd	When set, downgrades shareable exclusive load to shareable load when sending on CCIX Link 1 NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	1'b0
8	lnk1_dis_cpuevent_prop	When set, disables the propagation of CPU Events on CCIX Link 1 NOTE: This field is applicable only when SMP Mode enable parameter is set.	RW	1'b0
7:4	lnk1_num_snpercds	Controls the number of CCIX snoop credits assigned to Link 1 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
3	lnk1_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for CCIX Link 1	RW	1'b0

Table 3-166 por_cxg_ra_por_cxg_ra_cxpctl_link1_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	lnk1_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk1_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk1_link_en	<p>Enables CCIX Link 1 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_cxg_ra_cxpctl_link1_status

Functions as the CXRA CCIX Protocol Link 1 status register. Works with por_cxg_ra_cxpctl_link1_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C18

Register reset 64'b0010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-147 por_cxg_ra_por_cxg_ra_cxpctl_link1_status (high)

The following table shows the por_cxg_ra_cxpctl_link1_status higher register bit assignments.

Table 3-167 por_cxg_ra_por_cxg_ra_cxprtcl_link1_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

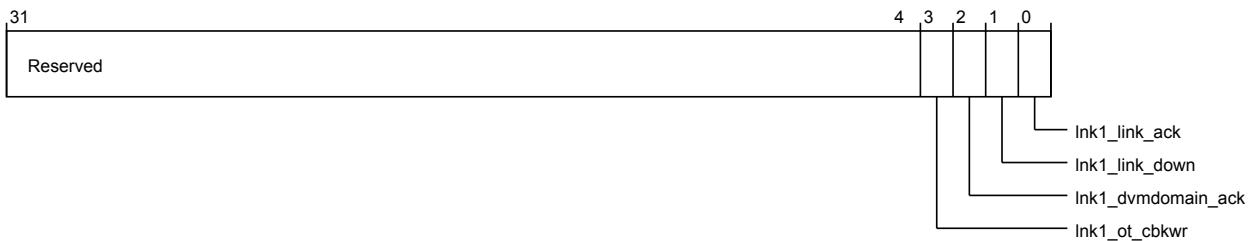


Figure 3-148 por_cxg_ra_por_cxg_ra_cxprtcl_link1_status (low)

The following table shows the por_cxg_ra_cxprtcl_link1_status lower register bit assignments.

Table 3-168 por_cxg_ra_por_cxg_ra_cxprtcl_link1_status (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	lnk1_ot_cbkwr	Provides status for outstanding CopyBack Write for CCIX Link1	RO	1'b0
2	lnk1_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for CCIX Link 1	RO	1'b0
1	lnk1_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	lnk1_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_cxg_ra_cxprtcl_link2_ctl

Functions as the CXRA CCIX Protocol Link 2 control register. Works with por_cxg_ra_cxprtcl_link2_status.

Its characteristics are:

Type	RW
Register width (Bits)	64

Address offset	16'h1C20
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

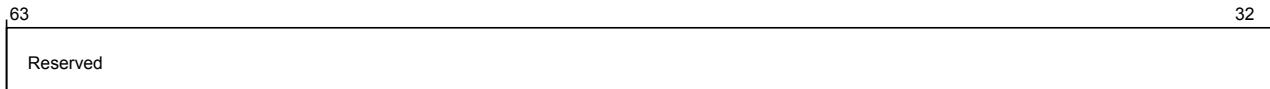


Figure 3-149 por_cxg_ra_por_cxg_ra_cxpctl_link2_ctl (high)

The following table shows the por_cxg_ra_cxpctl_link2_ctl higher register bit assignments.

Table 3-169 por_cxg_ra_por_cxg_ra_cxpctl_link2_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

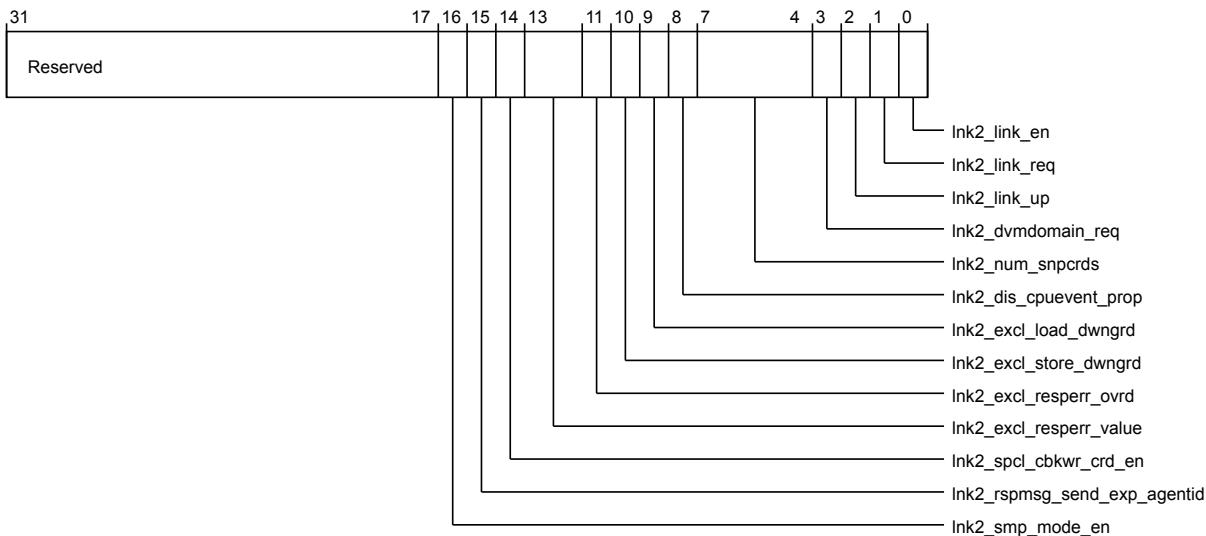


Figure 3-150 por_cxg_ra_por_cxg_ra_cxpctl_link2_ctl (low)

The following table shows the por_cxg_ra_cxpctl_link2_ctl lower register bit assignments.

Table 3-170 por_cxg_ra_por_cxg_ra_cxpctl_link2_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	lnk2_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 2.	RW	Configuration dependent

Table 3-170 por_cxg_ra_por_cxg_ra_cxprtl_link2_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
15	lnk2_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 2	RW	1'b0
14	lnk2_spcl_cbkwr_crd_en	When set, notifies RA to use special credits from HA to send CopyBack writes on CCIX Link 2 NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
13:12	lnk2_excl_resperr_value	Two bit value to override RespErr field of an exclusive response. Applicable only if lnk2_excl_resperr_ovrd bit is set. NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	2'b0
11	lnk2_excl_resperr_ovrd	When set, overrides the RespErr field of exclusive response with the lnk2_excl_resperr_value field NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	1'b0
10	lnk2_excl_store_dwngrd	When set, downgrades shareable exclusive store to shareable store when sending on CCIX Link 2 NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	1'b0
9	lnk2_excl_load_dwngrd	When set, downgrades shareable exclusive load to shareable load when sending on CCIX Link 2 NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring	RW	1'b0
8	lnk2_dis_cpuevent_prop	When set, disables the propagation of CPU Events on CCIX Link 2 NOTE: This field is applicable only when SMP Mode enable parameter is set.	RW	1'b0
7:4	lnk2_num_snpcrds	Controls the number of CCIX snoop credits assigned to Link 2 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
3	lnk2_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for CCIX Link 2	RW	1'b0

Table 3-170 por_cxg_ra_por_cxg_ra_cxpctl_link2_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	lnk2_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk2_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk2_link_en	<p>Enables CCIX Link 2 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_cxg_ra_cxpctl_link2_status

Functions as the CXRA CCIX Protocol Link 2 status register. Works with por_cxg_ra_cxpctl_link2_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C28

Register reset 64'b0010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-151 por_cxg_ra_por_cxg_ra_cxpctl_link2_status (high)

The following table shows the por_cxg_ra_cxpctl_link2_status higher register bit assignments.

Table 3-171 por_cxg_ra_por_cxg_ra_cxprtcl_link2_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

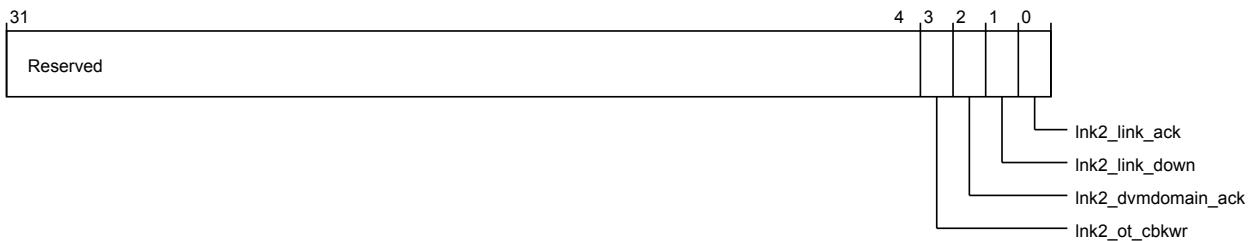


Figure 3-152 por_cxg_ra_por_cxg_ra_cxprtcl_link2_status (low)

The following table shows the por_cxg_ra_por_cxg_ra_cxprtcl_link2_status lower register bit assignments.

Table 3-172 por_cxg_ra_por_cxg_ra_cxprtcl_link2_status (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	Ink2_ot_cbkwr	Provides status for outstanding CopyBack Write for CCIX Link2	RO	1'b0
2	Ink2_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for CCIX Link 2	RO	1'b0
1	Ink2_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	Ink2_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

3.3.4 HN-I register descriptions

This section lists the HN-I registers.

por_hni_node_info

Provides component identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h0
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

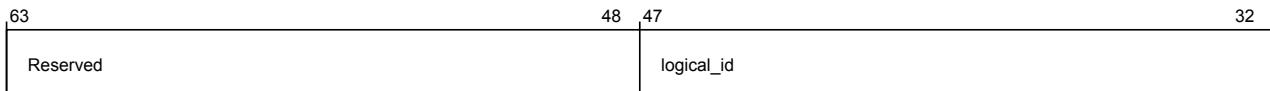


Figure 3-153 por_hni_por_hni_node_info (high)

The following table shows the por_hni_node_info higher register bit assignments.

Table 3-173 por_hni_por_hni_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

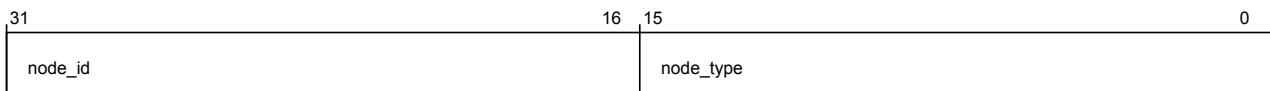


Figure 3-154 por_hni_por_hni_node_info (low)

The following table shows the por_hni_node_info lower register bit assignments.

Table 3-174 por_hni_por_hni_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	Configuration dependent

por_hni_child_info

Provides component child identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

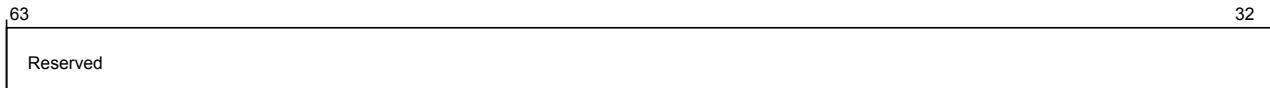


Figure 3-155 por_hni_por_hni_child_info (high)

The following table shows the por_hni_child_info higher register bit assignments.

Table 3-175 por_hni_por_hni_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

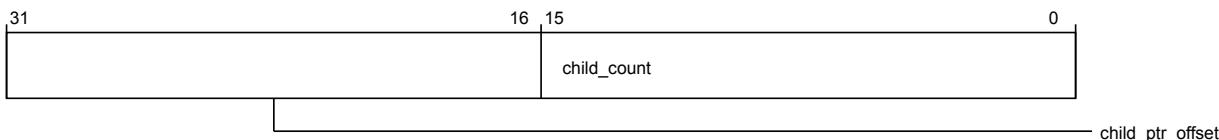


Figure 3-156 por_hni_por_hni_child_info (low)

The following table shows the por_hni_child_info lower register bit assignments.

Table 3-176 por_hni_por_hni_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_hni_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63
Reserved

32

Figure 3-157 por_hni_por_hni_secure_register_groups_override (high)

The following table shows the por_hni_secure_register_groups_override higher register bit assignments.

Table 3-177 por_hni_por_hni_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

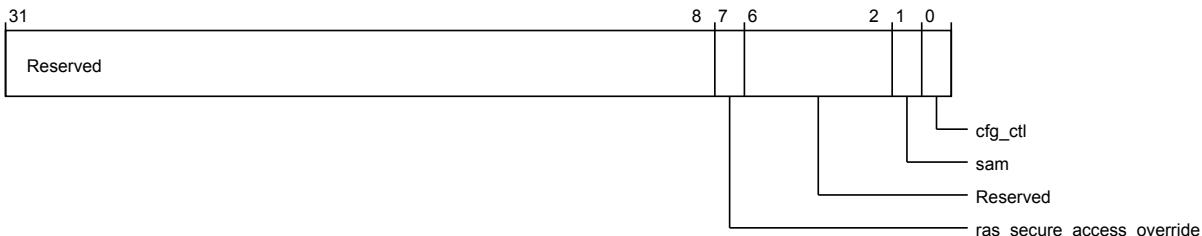


Figure 3-158 por_hni_por_hni_secure_register_groups_override (low)

The following table shows the por_hni_secure_register_groups_override lower register bit assignments.

Table 3-178 por_hni_por_hni_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0
6:2	Reserved	Reserved	RO	-
1	sam	Allows non-secure access to secure SAM registers	RW	1'b0
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_hni_unit_info

Provides component identification information for HN-I.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

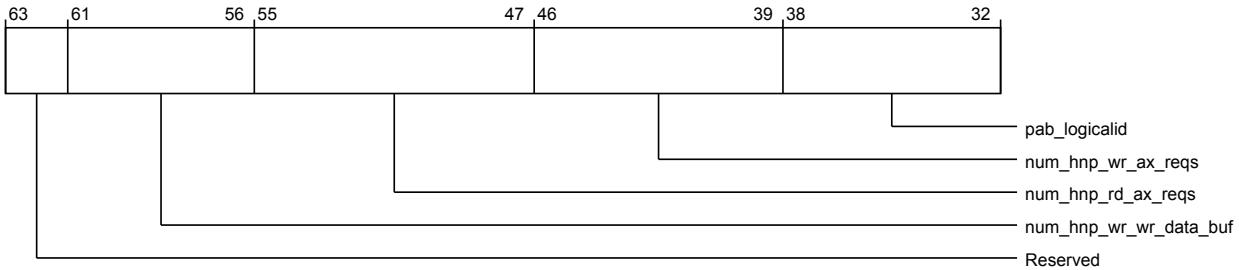


Figure 3-159 por_hni_por_hni_unit_info (high)

The following table shows the por_hni_unit_info higher register bit assignments.

Table 3-179 por_hni_por_hni_unit_info (high)

Bits	Field name	Description	Type	Reset
63:62	Reserved	Reserved	RO	-
61:56	num_hnp_wr_wr_data_buf	Number of P2P write data buffers in HN-I. HNP only	RO	Configuration dependent
55:47	num_hnp_rd_ax_reqs	Maximum number of outstanding P2P Read ACE-Lite/AXI4 requests.HNP only	RO	Configuration dependent
46:39	num_hnp_wr_ax_reqs	Maximum number of outstanding P2P Write ACE-Lite/AXI4 requests. HNP only	RO	Configuration dependent
38:32	pab_logicalid	PUB AUB bridge Logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

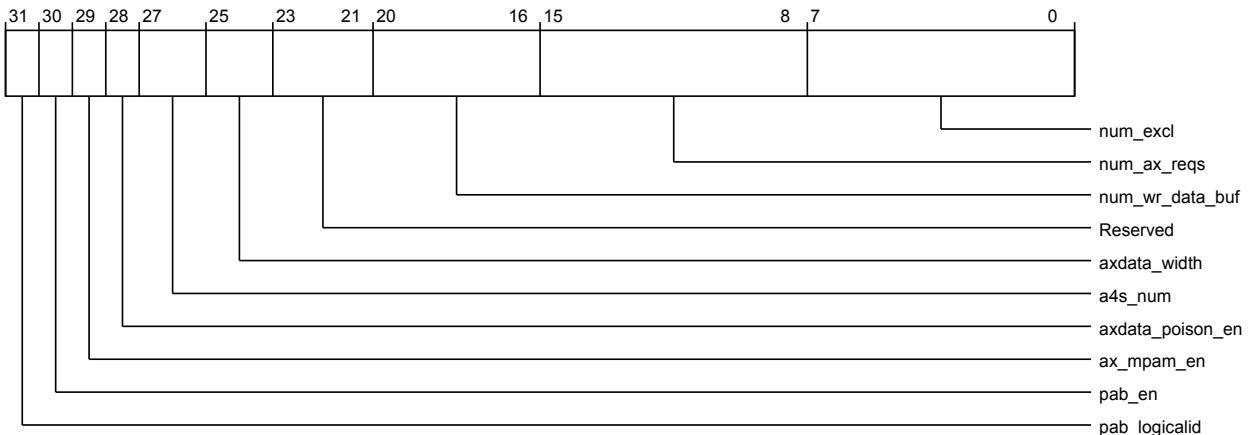


Figure 3-160 por_hni_por_hni_unit_info (low)

The following table shows the port number unit info lower register bit assignments.

Table 3-180 por_hni_por_hni_unit_info (low)

Bits	Field name	Description	Type	Reset
31	pab_logicalid	PUB AUB bridge Logical ID	RO	Configuration dependent
30	pab_en	PUB AUB bridge enable 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
29	ax_mpam_en	MPAM enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
28	axdata_poison_en	Data poison support on ACE-Lite/AXI4 interface 1'b0: Not supported 1'b1: Supported	RO	Configuration dependent
27:26	a4s_num	Number of AXI4Stream interfaces present	RO	Configuration dependent
25:24	axdata_width	Data width on ACE-Lite/AXI4 interface 2'b00: 128 bits 2'b01: 256 bits 2'b10: 512 bits	RO	Configuration dependent
23:21	Reserved	Reserved	RO	-
20:16	num_wr_data_buf	Number of write data buffers in HN-I	RO	Configuration dependent
15:8	num_ax_reqs	Maximum number of outstanding ACE-Lite/AXI4 requests	RO	Configuration dependent
7:0	num_excl	Number of exclusive monitors in HN-I	RO	Configuration dependent

por_hni_sam_addrregion0_cfg

Configures Address Region 0.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC00

Register reset 64'b1100011111

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hni_secure_register_groups_override.sam

The following image shows the higher register bit assignments.

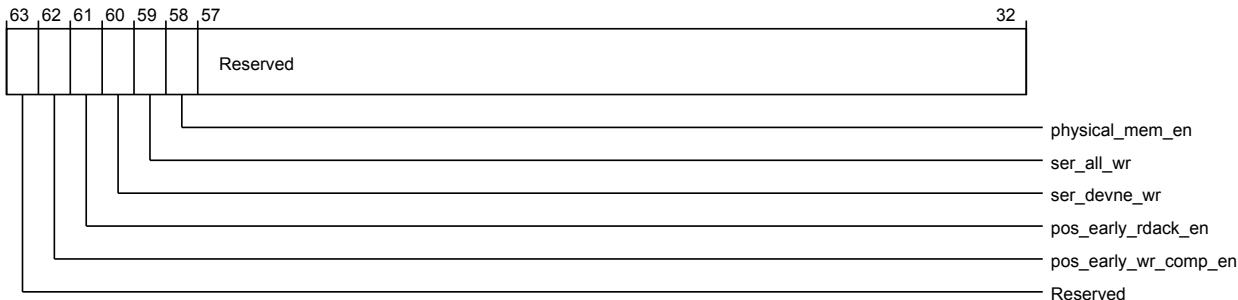


Figure 3-161 por_hni_por_hni_sam_addrregion0_cfg (high)

The following table shows the por_hni_sam_addrregion0_cfg higher register bit assignments.

Table 3-181 por_hni_por_hni_sam_addrregion0_cfg (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62	pos_early_wr_comp_en	Enables early write acknowledgment in Address Region 0; used to improve write performance	RW	1'b1
61	pos_early_rdack_en	Enables sending early read receipts from HN-I in Address Region 0; used to improve ordered read performance	RW	1'b1
60	ser_devne_wr	Used to serialize Device-nGnRnE writes within Address Region 0	RW	1'b0
59	ser_all_wr	Used to serialize all writes within Address Region 0	RW	1'b0
58	physical_mem_en	Address Region 0 follows Arm Architecture Reference Manual physical memory ordering guarantees	RW	1'b0
57:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

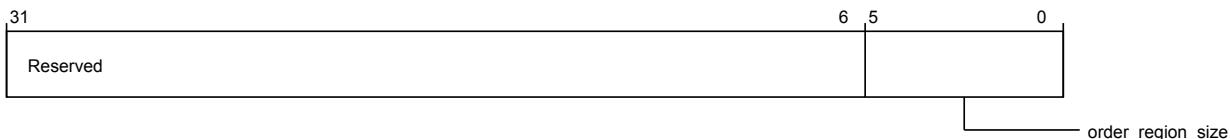


Figure 3-162 por_hni_por_hni_sam_addrregion0_cfg (low)

The following table shows the por_hni_sam_addrregion0_cfg lower register bit assignments.

Table 3-182 por_hni_por_hni_sam_addrregion0_cfg (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5:0	order_region_size	<n>; used to calculate Order Region 0 size within Address Region 0 (2^n*4KB)	RW	6'b111111

por_hni_sam_addrregion1_cfg

Configures Address Region 1.

Its characteristics are:

The following image shows the higher register bit assignments.

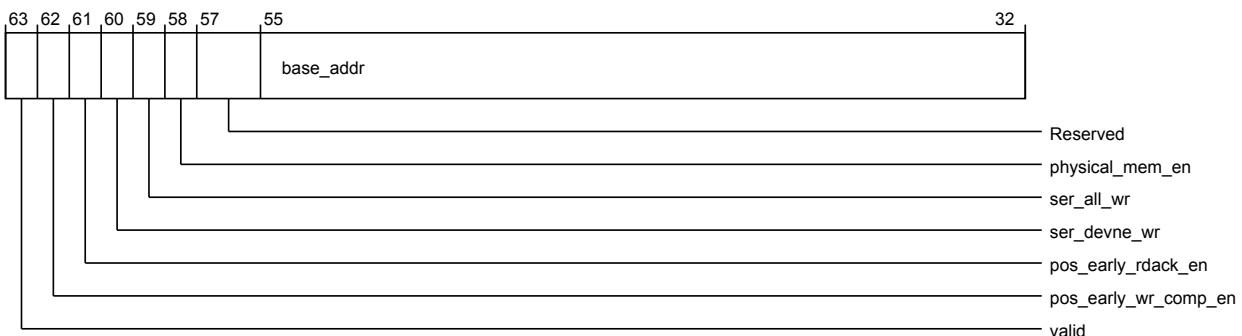


Figure 3-163 por_hni_por_hni_sam_addrregion1_cfg (high)

The following table shows the por_hni_sam_addrregion1_cfg higher register bit assignments.

Table 3-183 por_hni_por_hni_sam_addrregion1_cfg (high)

Bits	Field name	Description	Type	Reset
63	valid	Address Region 1 fields are programmed and valid	RW	1'h0
62	pos_early_wr_comp_en	Enables early write acknowledgment in Address Region 1; used to improve write performance	RW	1'b1
61	pos_early_rdack_en	Enables sending early read receipts from HN-I in Address Region 1; used to improve ordered read performance	RW	1'b1
60	ser_devne_wr	Used to serialize Device-nGnRnE writes within Address Region 1	RW	1'b0
59	ser_all_wr	Used to serialize all writes within Address Region 1	RW	1'b0
58	physical_mem_en	Address Region 1 follows Arm Architecture Reference Manual physical memory ordering guarantees	RW	1'b0
57:56	Reserved	Reserved	RO	-
55:32	base_addr	Address Region 1 base address; [address width-1:12] CONSTRAINT: Must be an integer multiple of the Address Region 1 size.	RW	40'h0

The following image shows the lower register bit assignments.

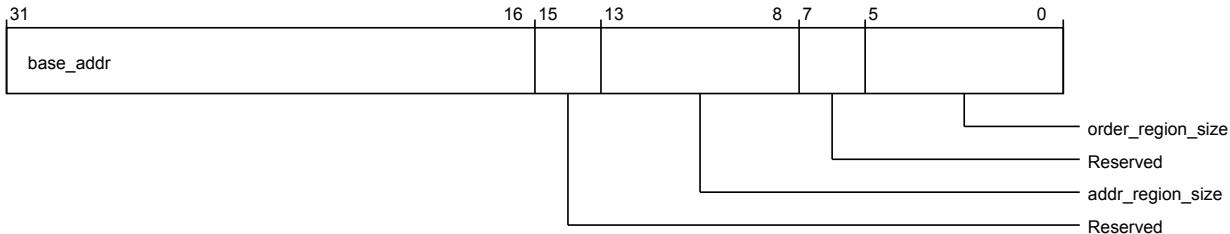


Figure 3-164 por_hni_por_hni_sam_addrregion1_cfg (low)

The following table shows the port_hni_sam_addrregion1_cfg lower register bit assignments.

Table 3-184 por_hni_por_hni_sam_addrregion1_cfg (low)

Bits	Field name	Description	Type	Reset
31:16	base_addr	Address Region 1 base address; [address width-1:12] CONSTRAINT: Must be an integer multiple of the Address Region 1 size.	RW	40'h0
15:14	Reserved	Reserved	RO	-
13:8	addr_region_size	<n>; used to calculate Address Region 1 size ($2^n * 4KB$) CONSTRAINT: <n> must be configured so that the Address Region 1 size is less than or equal to $2^{(address\ width)}$.	RW	6'h0
7:6	Reserved	Reserved	RO	-
5:0	order_region_size	<n>; used to calculate Order Region 1 size within Address Region 1 ($2^n * 4KB$)	RW	6'h0

por_hni_sam_addrregion2_cfg

Configures Address Region 2.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hni_secure_register_groups_override.sam

The following image shows the higher register bit assignments.

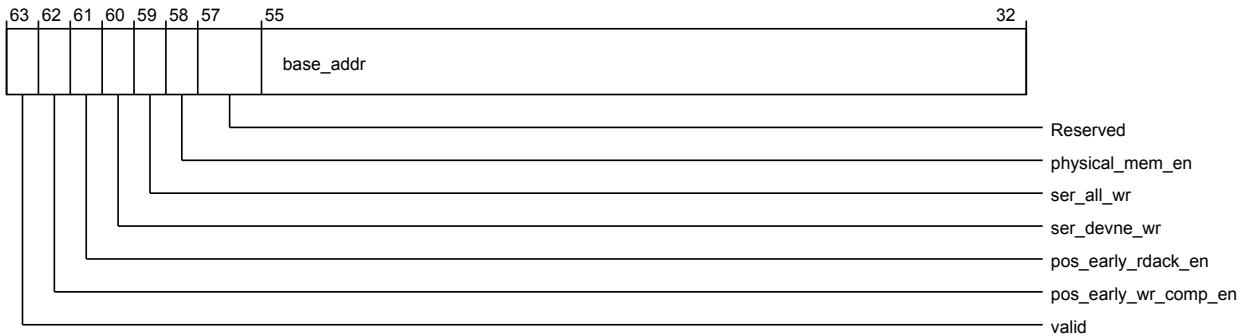


Figure 3-165 por_hni_por_hni_sam_addrregion2_cfg (high)

The following table shows the por_hni_sam_addrregion2_cfg higher register bit assignments.

Table 3-185 por_hni_por_hni_sam_addrregion2_cfg (high)

Bits	Field name	Description	Type	Reset
63	valid	Address Region 2 fields are programmed and valid	RW	1'h0
62	pos_early_wr_comp_en	Enables early write acknowledgment in Address Region 2; used to improve write performance	RW	1'b1
61	pos_early_rdack_en	Enables sending early read receipts from HN-I in Address Region 2; used to improve ordered read performance	RW	1'b1
60	ser_devne_wr	Used to serialize Device-nGnRnE writes within Address Region 2	RW	1'b0
59	ser_all_wr	Used to serialize all writes within Address Region 2	RW	1'b0
58	physical_mem_en	Address Region 2 follows Arm Architecture Reference Manual physical memory ordering guarantees	RW	1'b0
57:56	Reserved	Reserved	RO	-
55:32	base_addr	Address Region 2 base address; [address width-1:12] CONSTRAINT: Must be an integer multiple of the Address Region 2 size	RW	40'h0

The following image shows the lower register bit assignments.

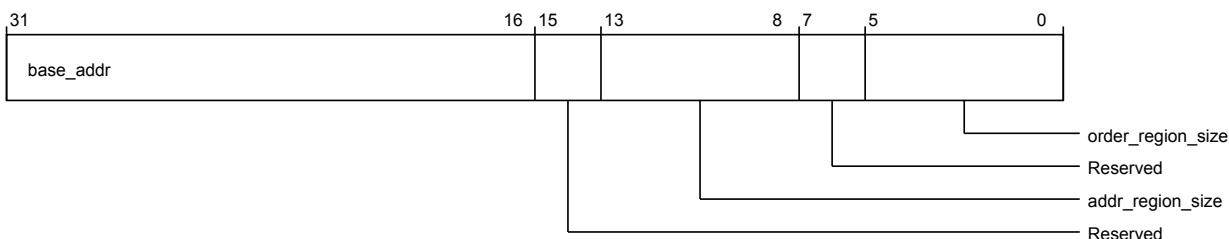


Figure 3-166 por_hni_por_hni_sam_addrregion2_cfg (low)

The following table shows the por_hni_sam_addrregion2_cfg lower register bit assignments.

Table 3-186 por_hni_por_hni_sam_addrregion2_cfg (low)

Bits	Field name	Description	Type	Reset
31:16	base_addr	Address Region 2 base address; [address width-1:12] CONSTRAINT: Must be an integer multiple of the Address Region 2 size	RW	40'h0
15:14	Reserved	Reserved	RO	-
13:8	addr_region_size	<n>; used to calculate Address Region 2 size ($2^n * 4KB$) CONSTRAINT: <n> must be configured so that the Address Region 2 size is less than or equal to $2^{(address\ width)}$.	RW	6'h0
7:6	Reserved	Reserved	RO	-
5:0	order_region_size	<n>; used to calculate Order Region 2 size within Address Region 2 ($2^n * 4KB$)	RW	6'h0

por_hni_sam_addrregion3_cfg

Configures Address Region 3.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'

Register reset 64'b0110

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hni_secure_register_groups_override.sam

The following image shows the higher register bit assignments.

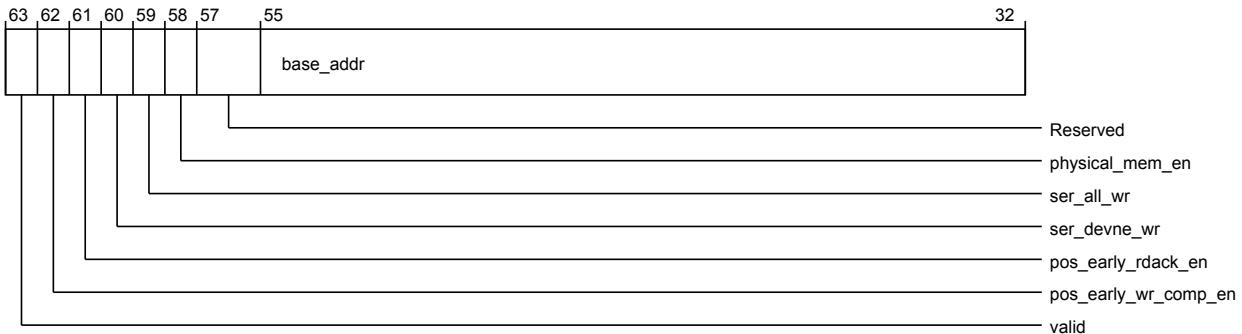


Figure 3-167 por_hni_por_hni_sam_addrregion3_cfg (high)

The following table shows the por_hni_sam_addrregion3_cfg higher register bit assignments.

Table 3-187 por_hni_por_hni_sam_addrregion3_cfg (high)

Bits	Field name	Description	Type	Reset
63	valid	Fields of Address Region 3 are programmed and valid	RW	1'h0
62	pos_early_wr_comp_en	Enables early write acknowledgment in Address Region 3; used to improve write performance	RW	1'b1
61	pos_early_rdack_en	Enables sending early read receipts from HN-I in Address Region 3; used to improve ordered read performance	RW	1'b1
60	ser_devne_wr	Used to serialize Device-nGnRnE writes within Address Region 3	RW	1'b0
59	ser_all_wr	Used to serialize all writes within Address Region 3	RW	1'b0
58	physical_mem_en	Address Region 3 follows Arm Architecture Reference Manual physical memory ordering guarantees	RW	1'b0
57:56	Reserved	Reserved	RO	-
55:32	base_addr	Address Region 3 base address; [address width-1:12] CONSTRAINT: Must be an integer multiple of the Address Region 3 size	RW	40'h0

The following image shows the lower register bit assignments.

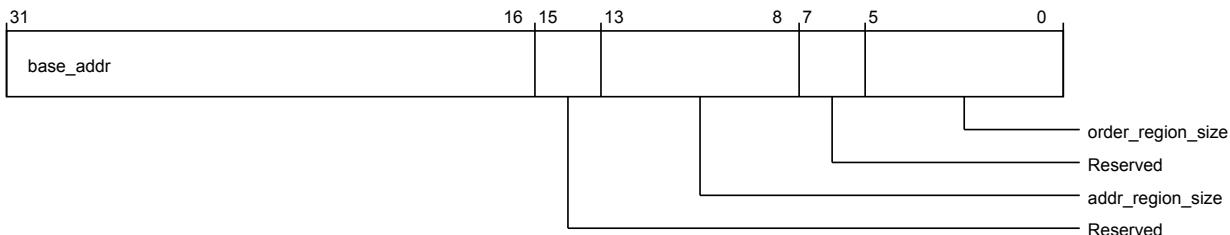


Figure 3-168 por_hni_por_hni_sam_addrregion3_cfg (low)

The following table shows the por_hni_sam_addrregion3_cfg lower register bit assignments.

Table 3-188 por_hni_por_hni_sam_addrregion3_cfg (low)

Bits	Field name	Description	Type	Reset
31:16	base_addr	Address Region 3 base address; [address width-1:12] CONSTRAINT: Must be an integer multiple of the Address Region 3 size	RW	40'h0
15:14	Reserved	Reserved	RO	-
13:8	addr_region_size	<n>; used to calculate Address Region 3 size ($2^n \times 4KB$) CONSTRAINT: <n> must be configured so that the Address Region 3 size is less than or equal to $2^{(address\ width)}$.	RW	6'h0
7:6	Reserved	Reserved	RO	-
5:0	order_region_size	<n>; used to calculate Order Region 3 size within Address Region 3 ($2^n \times 4KB$)	RW	6'h0

por_hni_cfg_ctl

Functions as the configuration control register for HN-I.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b001
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hni_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

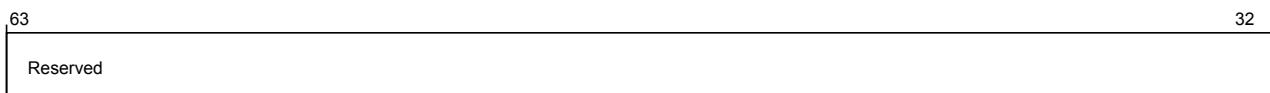


Figure 3-169 por_hni_por_hni_cfg_ctl (high)

The following table shows the por_hni_cfg_ctl higher register bit assignments.

Table 3-189 por_hni_por_hni_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

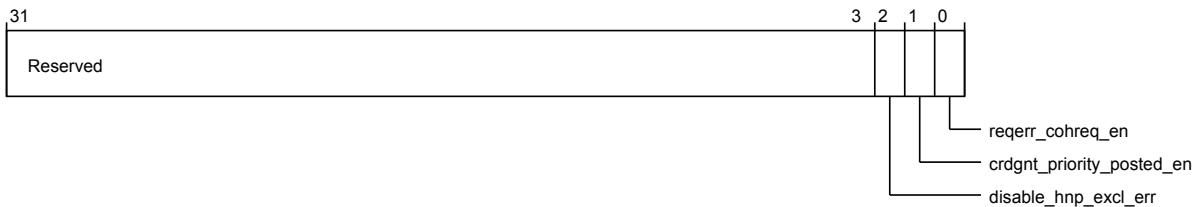


Figure 3-170 por_hni_por_hni_cfg_ctl (low)

The following table shows the por_hni_cfg_ctl lower register bit assignments.

Table 3-190 por_hni_por_hni_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	disable_hnp_excl_err	Disables sending NDE and Error logging on ReadNoSnp and WriteNoSnp Exclusives	RW	1'b0

Table 3-190 por_hni_por_hni_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
1	crdgnt_priority_posted_en	Enables High priority Credit Grant responses to Posted requests	RW	1'b0
0	reqerr_cohreq_en	Enables sending of NDE response error to RN and logging of error information for the following requests: 1. Coherent Read 2. CleanUnique/MakeUnique 3. Coherent/CopyBack Write	RW	1'b1

por_hni_aux_ctl

Functions as the auxiliary control register for HN-I.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA08

Register reset 64'b0

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

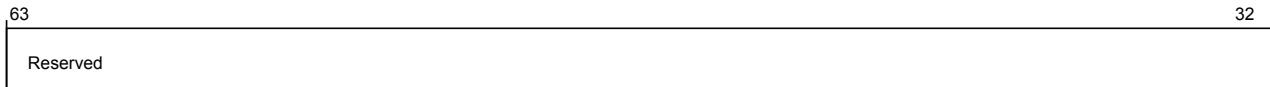


Figure 3-171 por_hni_por_hni_aux_ctl (high)

The following table shows the por_hni_aux_ctl higher register bit assignments.

Table 3-191 por_hni_por_hni_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

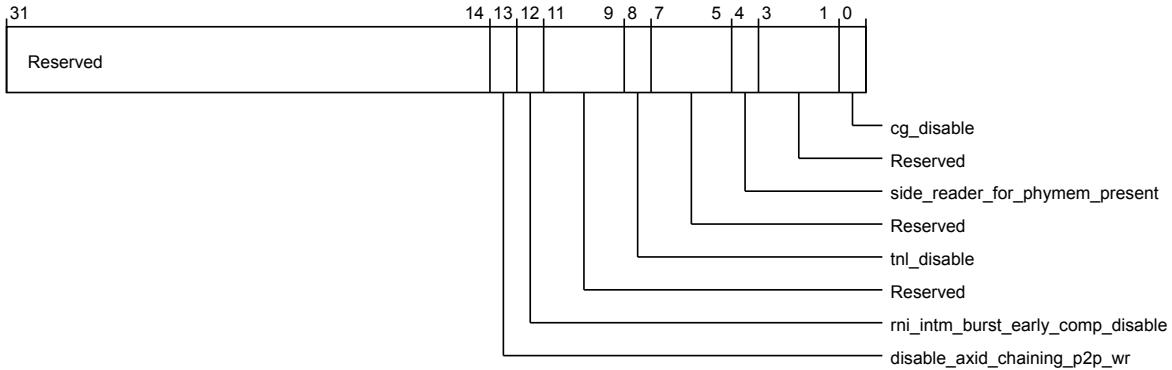


Figure 3-172 por_hni_por_hni_aux_ctl (low)

The following table shows the por_hni_aux_ctl lower register bit assignments.

Table 3-192 por_hni_por_hni_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:14	Reserved	Reserved	RO	-
13	disable_axid_chaining_p2p_wr	Disables AXID based chaining of PCIe writes in P2P Write slice. HNP only	RW	1'b0
12	rni_intm_burst_early_comp_disable	Disables Early COMP to RNI for non-last burst writes	RW	1'b0
11:9	Reserved	Reserved	RO	-
8	tnl_disable	Disables RNI-HNI Tunneling in HNI. por_rni_aux_ctl.dis_hni_wr_stream must be set before setting this bit	RW	1'b0
7:5	Reserved	Reserved	RO	-
4	side_reader_for_phymem_present	Enables side reader in physical memory range	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	cg_disable	Disables HN-I architectural clock gates	RW	1'b0

por_hni_errfr

Functions as the error feature register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3000

Register reset 64'b0000010100101

Usage constraints	Only accessible by secure accesses.
Secure group	por_hni_secure_register_groups_override.ras_secure_access_override

override

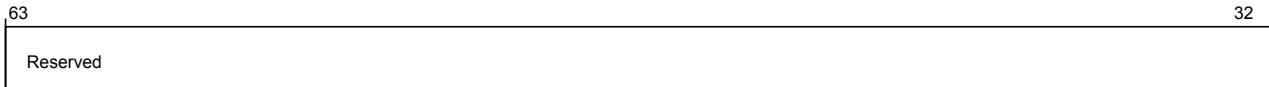


Figure 3-173 por_hni_por_hni_errfr (high)

The following table shows the por_hni_errfr higher register bit assignments.

Table 3-193 por_hni_por_hni_errfr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

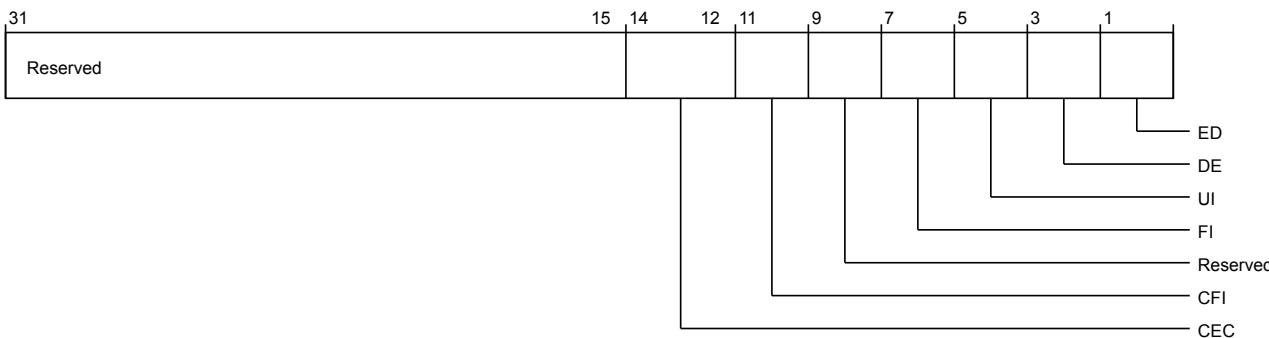


Figure 3-174 por_hni_por_hni_errfr (low)

The following table shows the por_hni_errfr lower register bit assignments.

Table 3-194 por_hni_por_hni_errfr (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_hni_errctlr

Functions as the error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3008
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group	por_hni_secure_register_groups_override.ras_secure_access_override override

The following image shows the higher register bit assignments.



Figure 3-175 por_hni_por_hni_errctlr (high)

The following table shows the por_hni_errctlr higher register bit assignments.

Table 3-195 por_hni_por_hni_errctlr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

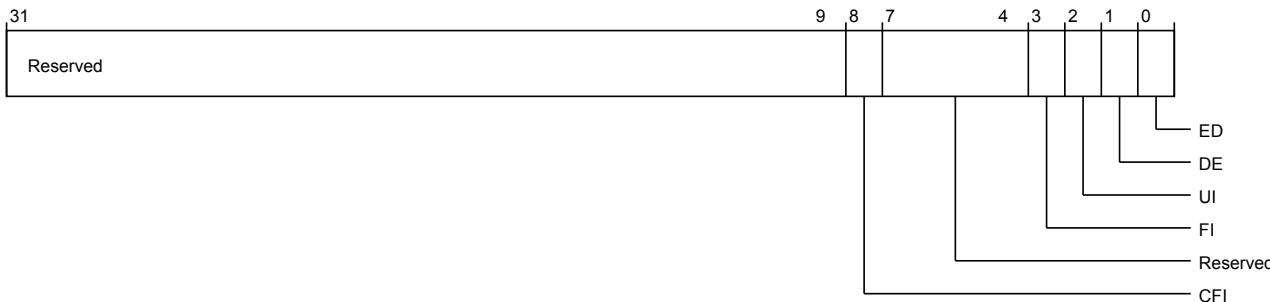


Figure 3-176 por_hni_por_hni_errctlr (low)

The following table shows the por_hni_errctlr lower register bit assignments.

Table 3-196 por_hni_por_hni_errctlr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_hni_errfr.CFI	RW	1'b0

Table 3-196 por_hni_por_hni_errctlr (low) (continued)

Bits	Field name	Description	Type	Reset
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_hni_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_hni_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_hni_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_hni_errfr.ED	RW	1'b0

por_hni_errstatus

Functions as the error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3010
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group	por_hni_secure_register_groups_override.ras_secure_access_override
override	

The following image shows the higher register bit assignments.

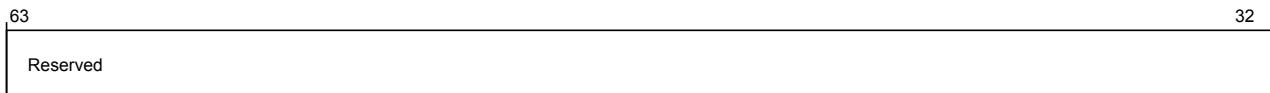


Figure 3-177 por_hni_por_hni_errstatus (high)

The following table shows the por_hni_errstatus higher register bit assignments.

Table 3-197 por_hni_por_hni_errstatus (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

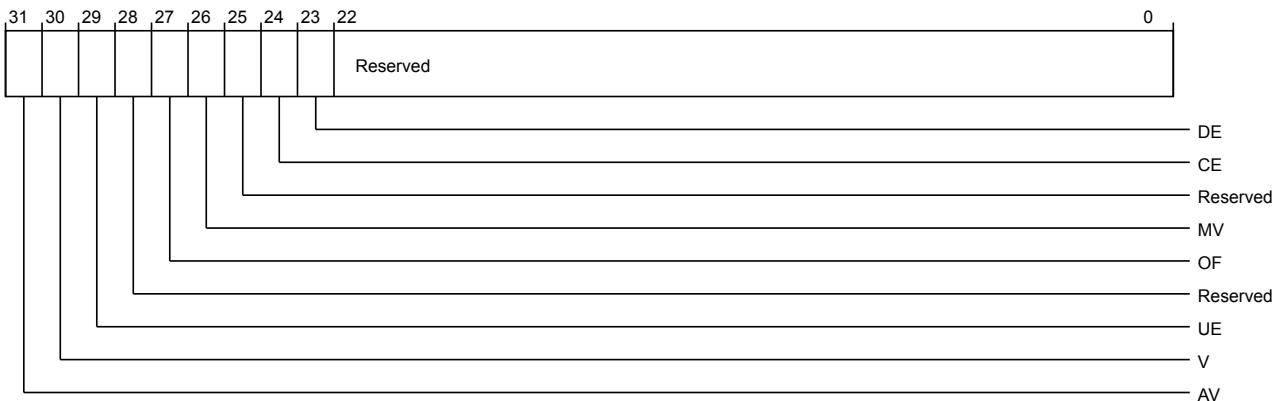


Figure 3-178 por_hni_por_hni_errstatus (low)

The following table shows the por_hni_errstatus lower register bit assignments.

Table 3-198 por_hni_por_hni_errstatus (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_hni_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_hni_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-

Table 3-198 por_hni_por_hni_errstatus (low) (continued)

Bits	Field name	Description	Type	Reset
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_hni_erraddr

Contains the error record address.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3018
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hni_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

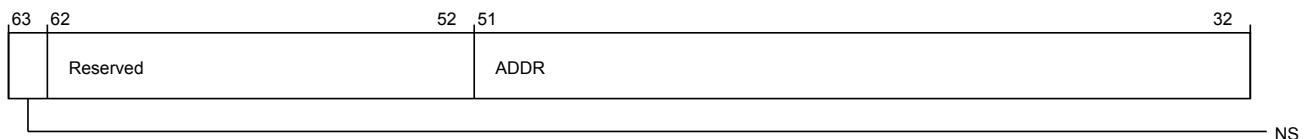


Figure 3-179 por_hni_por_hni_erraddr (high)

The following table shows the por_hni_erraddr higher register bit assignments.

Table 3-199 por_hni_por_hni_erraddr (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_hni_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.

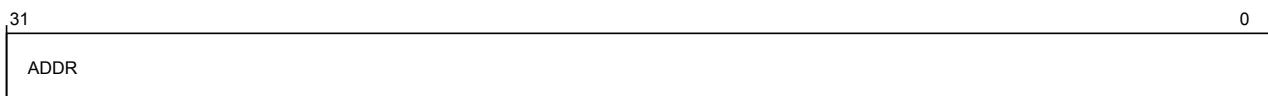


Figure 3-180 por_hni_por_hni_erraddr (low)

The following table shows the por_hni_erraddr lower register bit assignments.

Table 3-200 por_hni_por_hni_erraddr (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_hni_errmisc

Functions as the miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3020
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hni_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

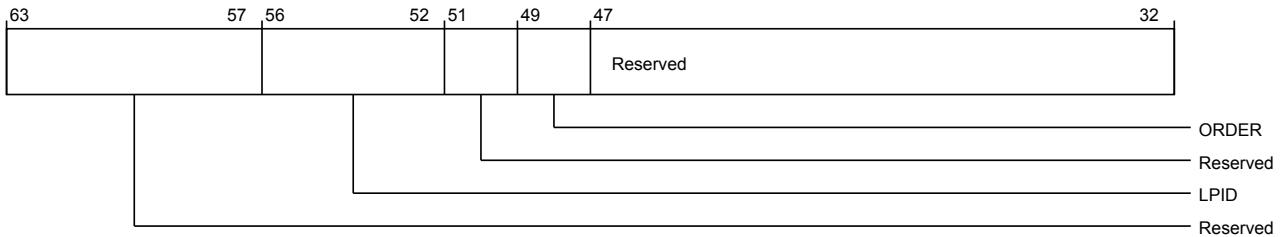


Figure 3-181 por_hni_por_hni_errmisc (high)

The following table shows the por_hni_errmisc higher register bit assignments.

Table 3-201 por_hni_por_hni_errmisc (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:52	LPID	Error logic processor ID	RW	5'b0
51:50	Reserved	Reserved	RO	-
49:48	ORDER	Error order	RW	4'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

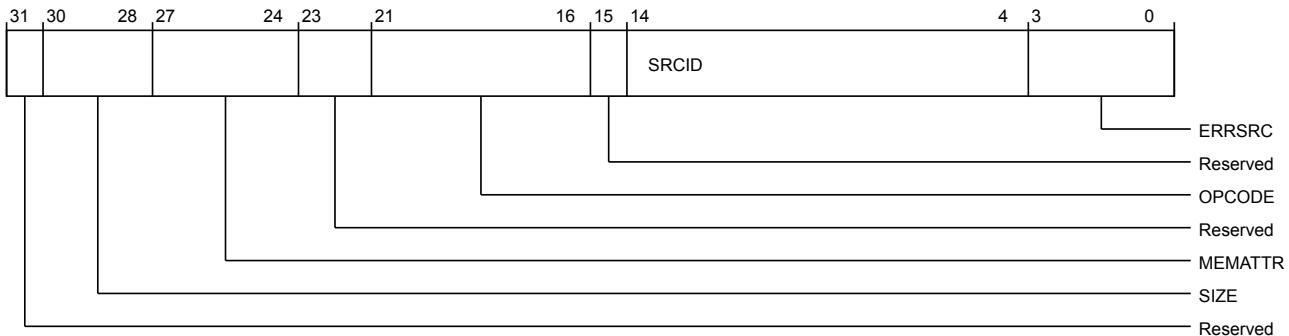


Figure 3-182 por_hni_por_hni_errmisc (low)

The following table shows the por_hni_errmisc lower register bit assignments.

Table 3-202 por_hni_por_hni_errmisc (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	SIZE	Error transaction size	RW	3'b0
27:24	MEMATTR	Error memory attributes	RW	4'b0
23:22	Reserved	Reserved	RO	-

Table 3-202 por_hni_por_hni_errmisc (low) (continued)

Bits	Field name	Description	Type	Reset
21:16	OPCODE	Error opcode	RW	6'b0
15	Reserved	Reserved	RO	-
14:4	SRCID	Error source ID	RW	11'b0
3:0	ERRSRC	Error source 4'b0000: Coherent read 4'b0001: Coherent write 4'b0010: CleanUnique/MakeUnique 4'b0011: Atomic 4'b0100: Illegal configuration read 4'b0101: Illegal configuration write 4'b0110: Configuration write data partial byte enable error 4'b0111: Configuration write data parity error or poison error 4'b1000: BRESP error 4'b1001: Poison error 4'b1010: BRESP error and poison error 4'b1011: Unsupported Exclusive access (HN-P only) NOTE: For configuration write data, BRESP, and poison errors, por_hni_errmisc.SRCID is the only valid field. For other error types, all fields are valid.	RW	4'b0

por_hni_errfr_NS

Functions as the non-secure error feature register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3100

Register reset 64'b00000010100101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

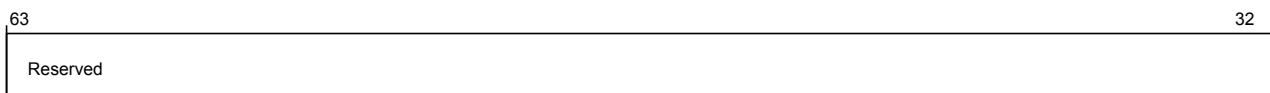


Figure 3-183 por_hni_por_hni_errfr_ns (high)

The following table shows the por_hni_errfr_NS higher register bit assignments.

Table 3-203 por_hni_por_hni_errfr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

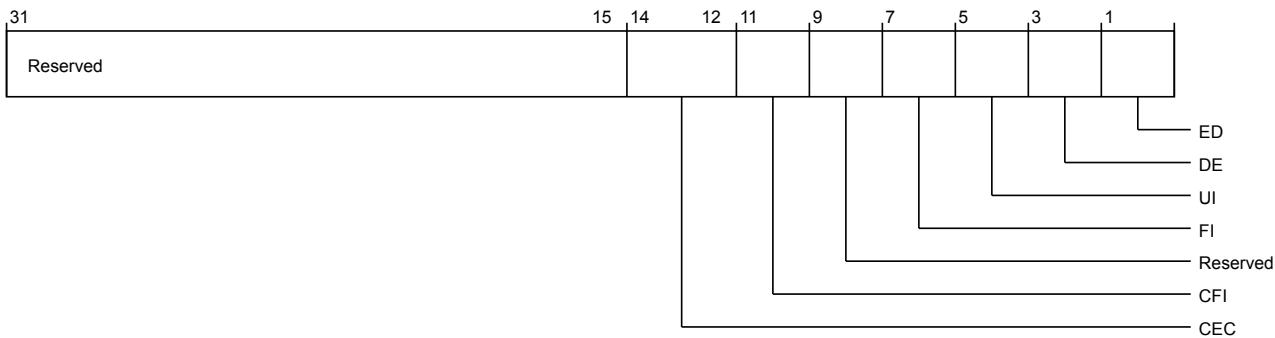


Figure 3-184 por_hni_por_hni_errfr_ns (low)

The following table shows the por_hni_errfr_NS lower register bit assignments.

Table 3-204 por_hni_por_hni_errfr_ns (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_hni_errctlr_NS

Functions as the non-secure error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3108
Register reset	64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

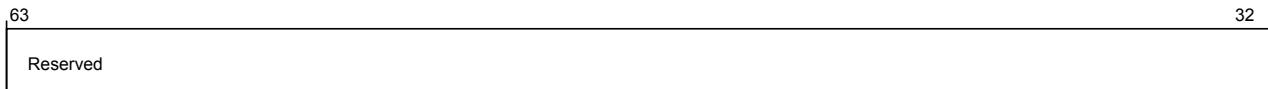


Figure 3-185 por_hni_por_hni_errctlr_ns (high)

The following table shows the por_hni_errctlr_NS higher register bit assignments.

Table 3-205 por_hni_por_hni_errctlr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

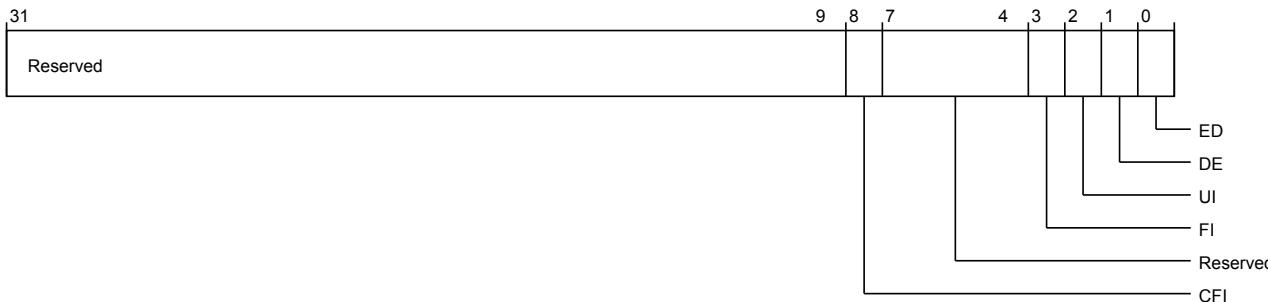


Figure 3-186 por_hni_por_hni_errctlr_ns (low)

The following table shows the por_hni_errctlr_NS lower register bit assignments.

Table 3-206 por_hni_por_hni_errctlr_ns (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_hni_errfr_NS.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_hni_errfr_NS.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_hni_errfr_NS.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_hni_errfr_NS.DE	RW	1'b0
0	ED	Enables error detection as specified in por_hni_errfr_NS.ED	RW	1'b0

por_hni_errstatus_NS

Functions as the non-secure error status register.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h3110

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-187 por_hni_por_hni_errstatus_ns (high)

The following table shows the por_hni_errstatus_NS higher register bit assignments.

Table 3-207 por_hni_por_hni_errstatus_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

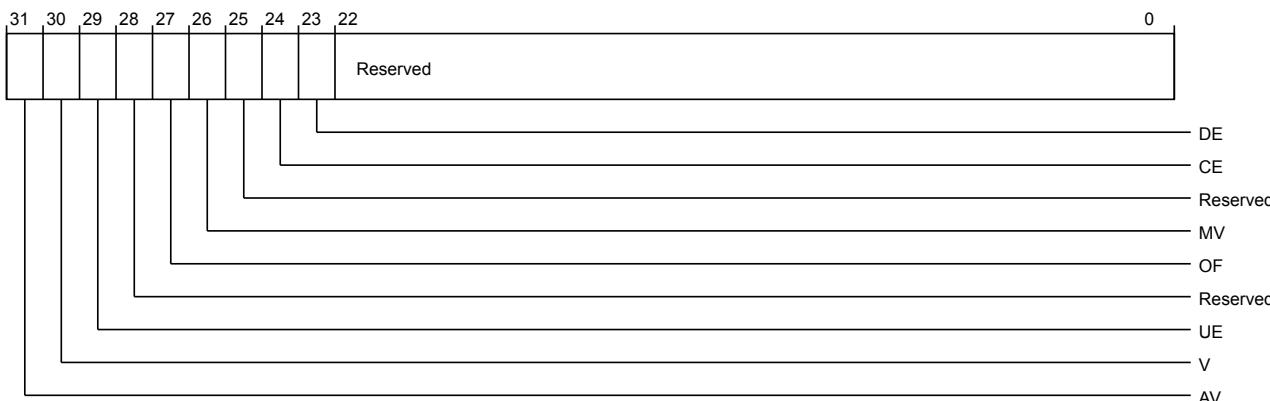


Figure 3-188 por_hni_por_hni_errstatus_ns (low)

The following table shows the por_hni_errstatus_NS lower register bit assignments.

Table 3-208 por_hni_por_hni_errstatus_ns (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_hni_erraddr_NS contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_hni_errmisc_NS valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_hni_erraddr_NS

Contains the non-secure error record address.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3118
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

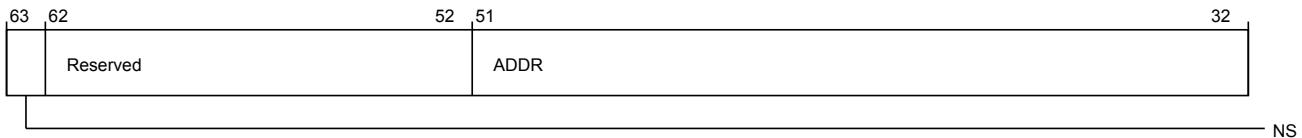


Figure 3-189 por_hni_por_hni_erraddr_ns (high)

The following table shows the por_hni_erraddr_NS higher register bit assignments.

Table 3-209 por_hni_por_hni_erraddr_ns (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_hni_erraddr_NS.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.

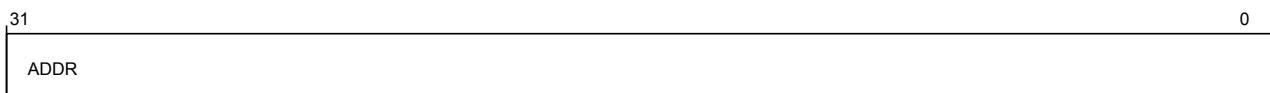


Figure 3-190 por_hni_por_hni_erraddr_ns (low)

The following table shows the por_hni_erraddr_NS lower register bit assignments.

Table 3-210 por_hni_por_hni_erraddr_ns (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_hni_errmisc_NS

Functions as the non-secure miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3120
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

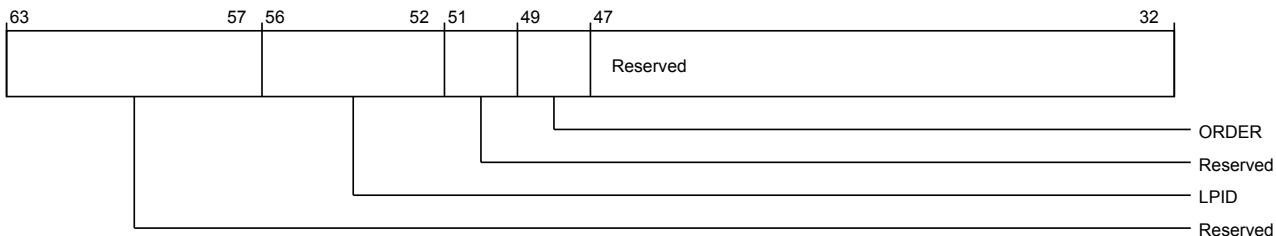


Figure 3-191 por_hni_por_hni_errmisc_ns (high)

The following table shows the por_hni_errmisc_NS higher register bit assignments.

Table 3-211 por_hni_por_hni_errmisc_ns (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:52	LPID	Error logic processor ID	RW	5'b0
51:50	Reserved	Reserved	RO	-
49:48	ORDER	Error order	RW	4'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

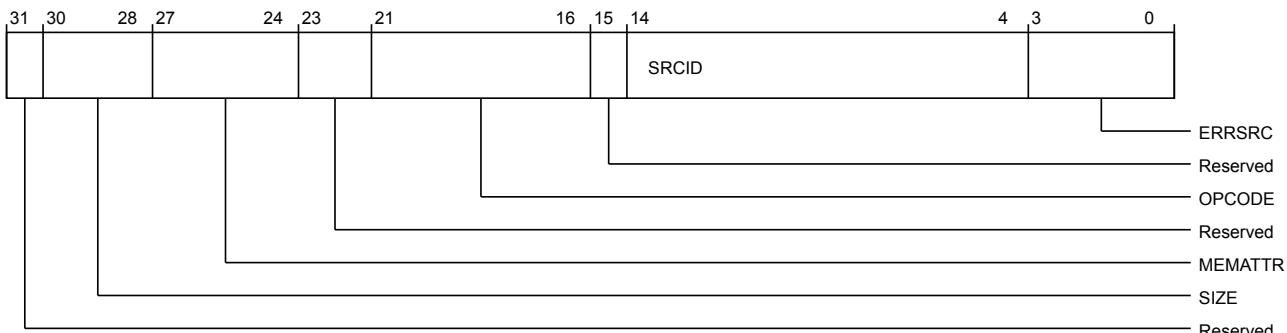


Figure 3-192 por_hni_por_hni_errmisc_ns (low)

The following table shows the por_hni_errmisc_NS lower register bit assignments.

Table 3-212 por_hni_por_hni_errmisc_ns (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	SIZE	Error transaction size	RW	3'b0
27:24	MEMATTR	Error memory attributes	RW	4'b0
23:22	Reserved	Reserved	RO	-
21:16	OPCODE	Error opcode	RW	6'b0
15	Reserved	Reserved	RO	-
14:4	SRCID	Error source ID	RW	11'b0
3:0	ERRSRC	Error source 4'b0000: Coherent read 4'b0001: Coherent write 4'b0010: CleanUnique/MakeUnique 4'b0011: Atomic 4'b0100: Illegal configuration read 4'b0101: Illegal configuration write 4'b0110: Configuration write data partial byte enable error 4'b0111: Configuration write data parity error or poison error 4'b1000: BRESP error 4'b1001: Poison error 4'b1010: BRESP error and poison error 4'b1011: Unsupported Exclusive access (HN-P only) NOTE: For configuration write data, BRESP, and poison errors, por_hni_errmisc_NS.SRCID is the only valid field. For other error types, all fields are valid.	RW	4'b0

por_hni_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2000
Register reset	64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

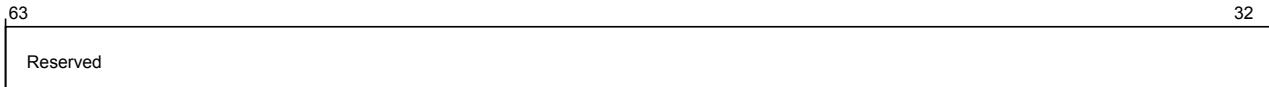


Figure 3-193 por_hni_por_hni_pmu_event_sel (high)

The following table shows the por_hni_pmu_event_sel higher register bit assignments.

Table 3-213 por_hni_por_hni_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

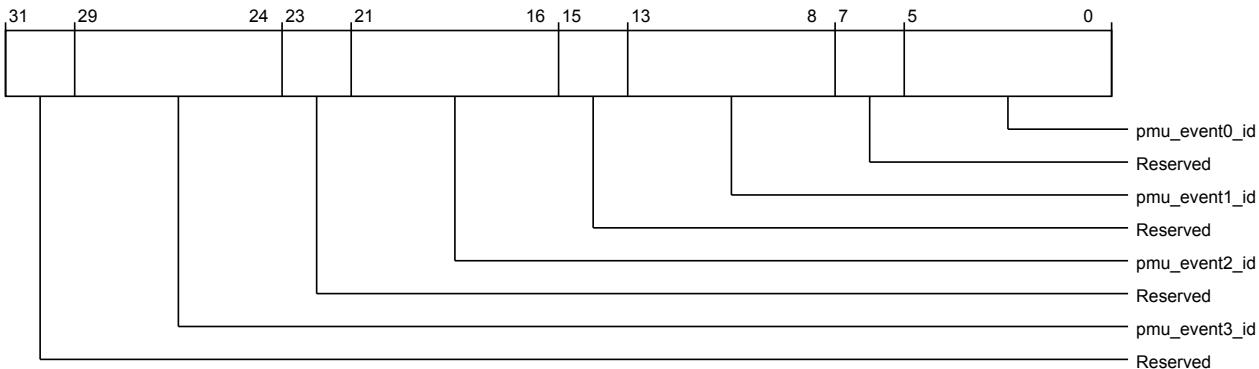


Figure 3-194 por_hni_por_hni_pmu_event_sel (low)

The following table shows the por_hni_pmu_event_sel lower register bit assignments.

Table 3-214 por_hni_por_hni_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	HN-I PMU Event 3 select; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	HN-I PMU Event 2 select; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	HN-I PMU Event 1 select; see pmu_event0_id for encodings	RW	6'b0

Table 3-214 por_hni_por_hni_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-
5:0	pmu_event0_id	HN-I PMU Event 0 select 6'h00: No event 6'h20: RRT read occupancy count overflow 6'h21: RRT write occupancy count overflow 6'h22: RDT read occupancy count overflow 6'h23: RDT write occupancy count overflow 6'h24: WDB occupancy count overflow 6'h25: RRT read allocation 6'h26: RRT write allocation 6'h27: RDT read allocation 6'h28: RDT write allocation 6'h29: WDB allocation 6'h2A: RETRYACK TXRSP flit sent 6'h2B: ARVALID set without ARREADY event 6'h2C: ARREADY set without ARVALID event 6'h2D: AWVALID set without AWREADY event 6'h2E: AWREADY set without AWVALID event 6'h2F: WVALID set without WREADY event 6'h30: TXDAT stall (TXDAT valid but no link credit available) 6'h31: Non-PCIe serialization event 6'h32: PCIe serialization event NOTE: All other encodings are reserved.	RW	6'b0

por_hnp_pmu_event_sel

Specifies the PMU event to be counted. HNP only

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2008

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

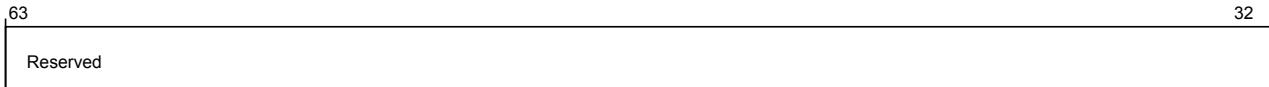


Figure 3-195 por_hni_por_hnp_pmu_event_sel (high)

The following table shows the por_hnp_pmu_event_sel higher register bit assignments.

Table 3-215 por_hni_por_hnp_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

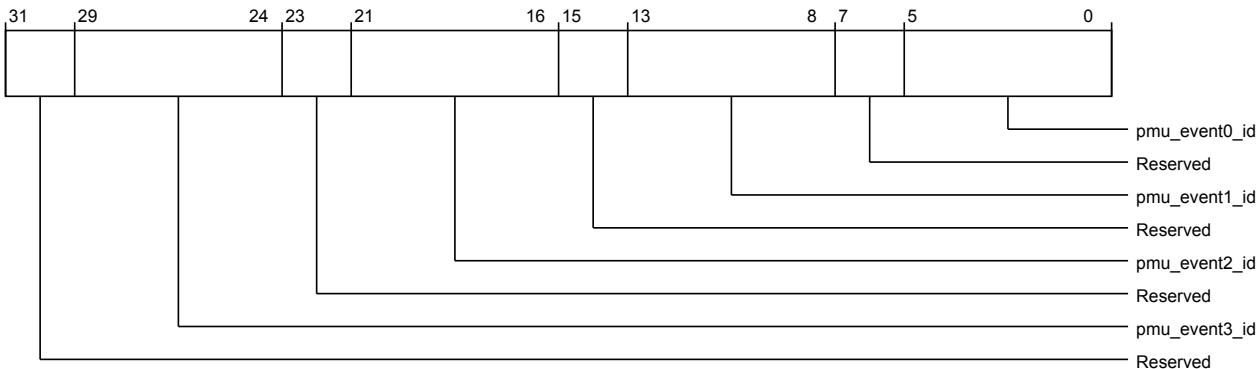


Figure 3-196 por_hni_por_hnp_pmu_event_sel (low)

The following table shows the por_hnp_pmu_event_sel lower register bit assignments.

Table 3-216 por_hni_por_hnp_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	P2P Slice PMU Event 3 select; see pmu_event0_id for encodings"	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	P2P Slice PMU Event 2 select; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	P2P Slice PMU Event 1 select; see pmu_event0_id for encodings	RW	6'b0

Table 3-216 por_hni_por_hnp_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-
5:0	pmu_event0_id	P2P Slice PMU Event 0 select 6'h00: No event 6'h01: RRT write occupancy count overflow 6'h02: RDT write occupancy count overflow 6'h03: WDB occupancy count overflow 6'h04: RRT write allocation 6'h05: RDT write allocation 6'h06: WDB allocation 6'h07: AWVALID set without AWREADY event 6'h08: AWREADY set without AWVALID event 6'h09: WVALID set without WREADY event 6'h11: RRT read occupancy count overflow 6'h12: RDT read occupancy count overflow 6'h13: RRT read allocation 6'h14: RDT read allocation 6'h15: ARVALID set without ARREADY event 6'h16: ARREADY set without ARVALID event NOTE: All other encodings are reserved.	RW	6'b0

3.3.5 MTU register descriptions

This section lists the HN-I registers.

por_mtu_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	48 47	32
Reserved	logical_id	

Figure 3-197 por_mtu_por_mtu_node_info (high)

The following table shows the por_mtu_node_info higher register bit assignments.

Table 3-217 por_mtu_por_mtu_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

31	16 15	0
node_id	node_type	

Figure 3-198 por_mtu_por_mtu_node_info (low)

The following table shows the por_mtu_node_info lower register bit assignments.

Table 3-218 por_mtu_por_mtu_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0010

por_mtu_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

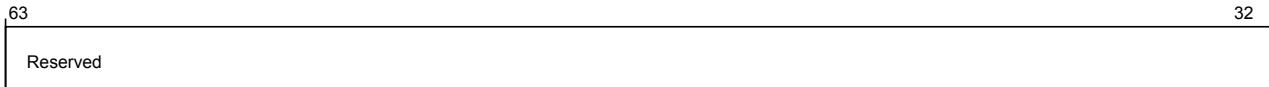


Figure 3-199 por_mtu_por_mtu_child_info (high)

The following table shows the por_mtu_child_info higher register bit assignments.

Table 3-219 por_mtu_por_mtu_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

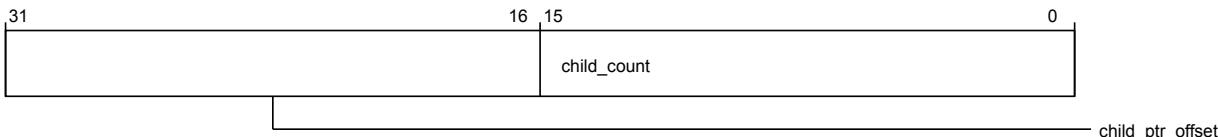


Figure 3-200 por_mtu_por_mtu_child_info (low)

The following table shows the por_mtu_child_info lower register bit assignments.

Table 3-220 por_mtu_por_mtu_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_mtu_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

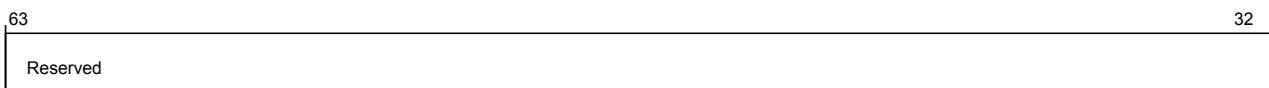


Figure 3-201 por_mtu_por_mtu_secure_register_groups_override (high)

The following table shows the por_mtu_secure_register_groups_override higher register bit assignments.

Table 3-221 por_mtu_por_mtu_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

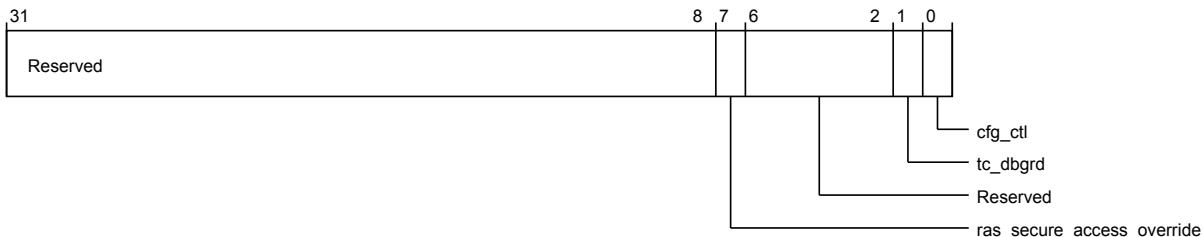


Figure 3-202 por_mtu_por_mtu_secure_register_groups_override (low)

The following table shows the por_mtu_secure_register_groups_override lower register bit assignments.

Table 3-222 por_mtu_por_mtu_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0
6:2	Reserved	Reserved	RO	-
1	tc_dbgrd	Allows non-secure access to secure debug register (por_mtu_cfg_tc_dgbrd)	RW	1'b0
0	cfg_ctl	Allows non-secure access to secure configuration control register (por_mtu_cfg_ctl)	RW	1'b0

por_mtu_unit_info

Provides component identification information for MTU.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

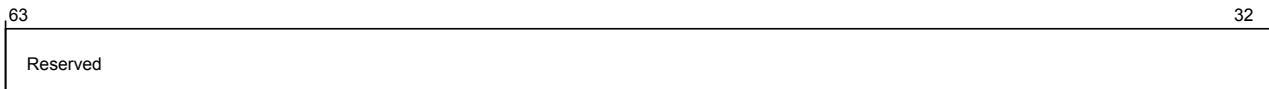


Figure 3-203 por_mtu_por_mtu_unit_info (high)

The following table shows the por_mtu_unit_info higher register bit assignments.

Table 3-223 por_mtu_por_mtu_unit_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

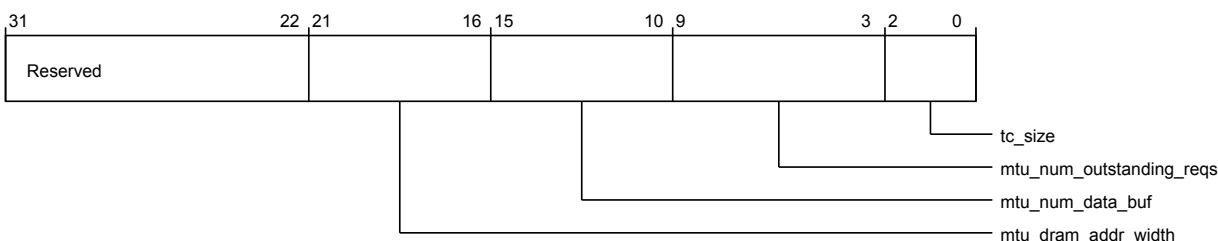


Figure 3-204 por_mtu_por_mtu_unit_info (low)

The following table shows the por_mtu_unit_info lower register bit assignments.

Table 3-224 por_mtu_por_mtu_unit_info (low)

Bits	Field name	Description	Type	Reset
31:22	Reserved	Reserved	RO	-
21:16	mtu_dram_addr_width	DRAM Addr width	RO	Configuration dependent
15:10	mtu_num_data_buf	Number of data buffers in MTU	RO	Configuration dependent
9:3	mtu_num_outstanding_reqs	Maximum number of outstanding AXI requests from MTU	RO	Configuration dependent
2:0	tc_size	TC size 3'b000: No TC 3'b001: 128KB 3'b010: 256KB 3'b011: 512KB 3'b100: 1MB 3'b101: 2MB	RO	-

por_mtu_cfg_ctl

Functions as the configuration control register for MTU.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mtu_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

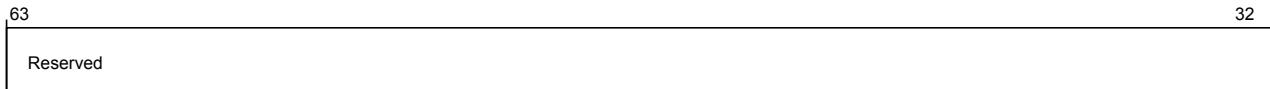


Figure 3-205 por_mtu_por_mtu_cfg_ctl (high)

The following table shows the por_mtu_cfg_ctl higher register bit assignments.

Table 3-225 por_mtu_por_mtu_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

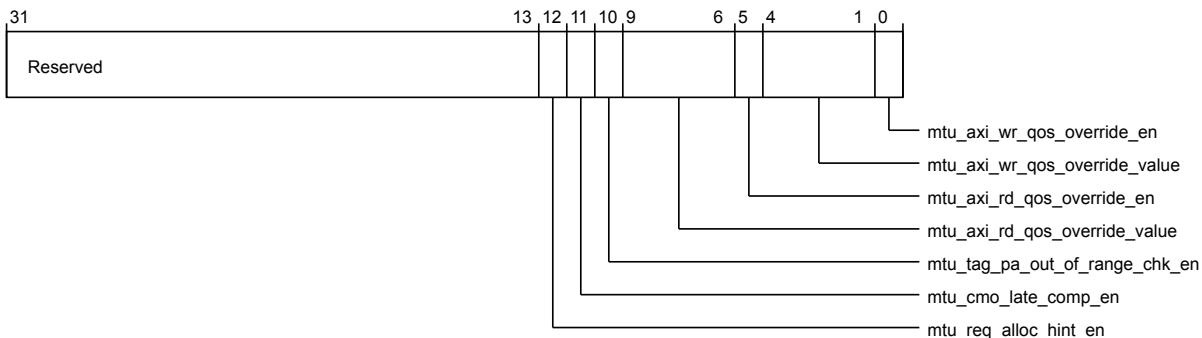


Figure 3-206 por_mtu_por_mtu_cfg_ctl (low)

The following table shows the por_mtu_cfg_ctl lower register bit assignments.

Table 3-226 por_mtu_por_mtu_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:13	Reserved	Reserved	RO	-
12	mtu_req_alloc_hint_en	Request Alloc Hint is used to determine TC allocation for TC Miss. Note: If this bit is clear, Tags are always allocated in TC. During no_tc_mode, and no_fill_mode; Tags are never allocated.	RW	1'b0

Table 3-226 por_mtu_por_mtu_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
11	mtu_cmo_late_comp_en	Enables CMO completion only when data is flushed out of TC to Memory	RW	1'b0
10	mtu_tag_pa_out_of_range_chk_en	Enables Tag Address Out of Range checking.	RW	1'b0
9:6	mtu_axi_rd_qos_override_value	QoS Override value to be used for all AXI Read requests generated by MTU. This value is used only when axi_rd_qos_override_en is set.	RW	4'b0000
5	mtu_axi_rd_qos_override_en	Enables QoS override for all AXI Read requests.	RW	1'b0
4:1	mtu_axi_wr_qos_override_value	QoS Override value to be used for all AXI Write requests generated by MTU. This value is used only when axi_wr_qos_override_en is set.	RW	4'b0000
0	mtu_axi_wr_qos_override_en	Enables QoS override for all AXI Write requests.	RW	1'b0

por_mtu_aux_ctl

Functions as the auxiliary control register for the MTU

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	64'b00001000
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

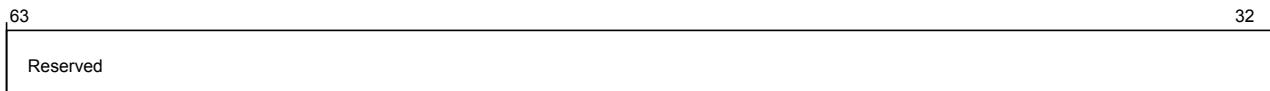


Figure 3-207 por_mtu_por_mtu_aux_ctl (high)

The following table shows the por_mtu_aux_ctl higher register bit assignments.

Table 3-227 por_mtu_por_mtu_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

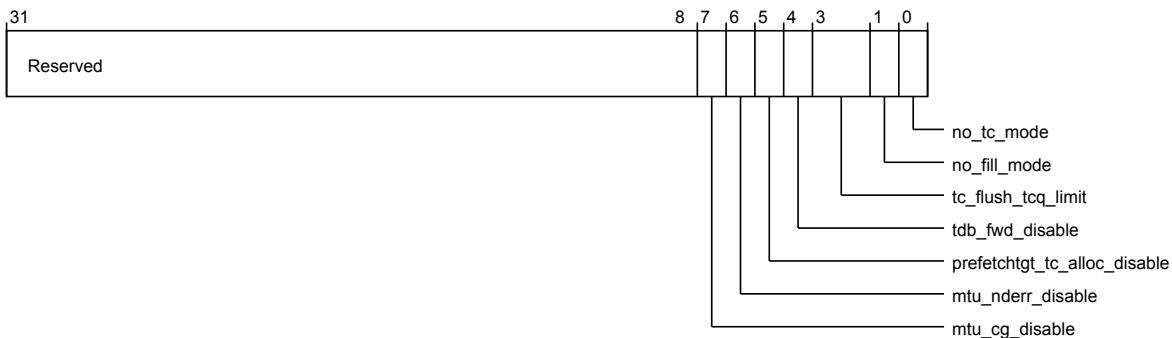


Figure 3-208 por_mtu_por_mtu_aux_ctl (low)

The following table shows the por_mtu_aux_ctl lower register bit assignments.

Table 3-228 por_mtu_por_mtu_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	mtu_cg_disable	Disables MTU architectural clock gating.	RW	1'b0
6	mtu_nderr_disable	Disable NDErr from MTU to RN for any Tag errors.	RW	1'b0
5	prefetchtgt_tc_alloc_disable	Do not allocate PrefetchTgt requests in TC.	RW	1'b0
4	tdb_fwd_disable	Disable Tag Data Buffer forwarding from a request to a dependent child request.	RW	1'b0
3:2	tc_flush_tcq_limit	Controls number of TC Flush requests allowed to occupy TCQ entries. 2'b00: TC Flush Requests allowed to take one TCQ entry. 2'b01: TC Flush Requests allowed to take 25% of TCQ entries. 2'b10: TC Flush Requests allowed to take 50% of TCQ entries. 2'b11: TC Flush Requests allowed to take All TCQ entries.	RW	2'b10
1	no_fill_mode	Enables No Fill Mode for Tag Cache. When set, no new lines would be allocated in Tag Cache.	RW	1'b0
0	no_tc_mode	Enables No TC Mode; disables MTU Tag Cache when set.	RW	1'b0

por_mtu_tc_flush_pr

Functions as Tag Cache Flush Policy Register

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA30

Register reset 64'b0

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

Secure group override por_mtu_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

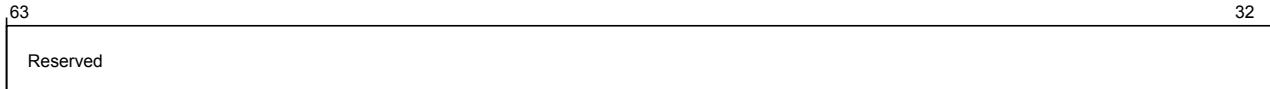


Figure 3-209 por_mtu_tc_flush_pr (high)

The following table shows the por_mtu_tc_flush_pr higher register bit assignments.

Table 3-229 por_mtu_tc_flush_pr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

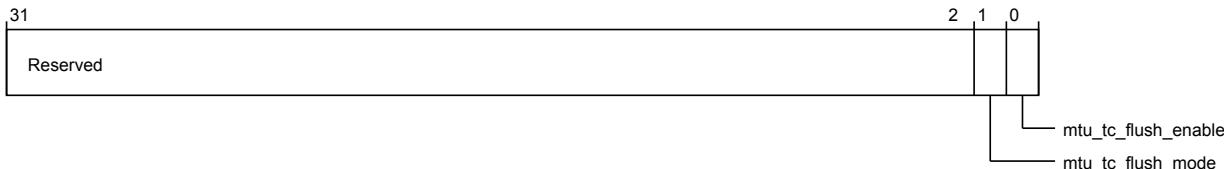


Figure 3-210 por_mtu_tc_flush_pr (low)

The following table shows the por_mtu_tc_flush_pr lower register bit assignments.

Table 3-230 por_mtu_tc_flush_pr (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1	mtu_tc_flush_mode	Tag Cache Flush Mode 1'b0: Clean Invalid. WB dirty data and invalidate local copy in TC. 1'b1: Clean Shared. WB dirty data and keep clean copy in TC.	RW	1'b0
0	mtu_tc_flush_enable	Start Tag Cache Flush	RW	1'b0

por_mtu_tc_flush_sr

Functions as Tag Cache Flush Status Register

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hA38

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

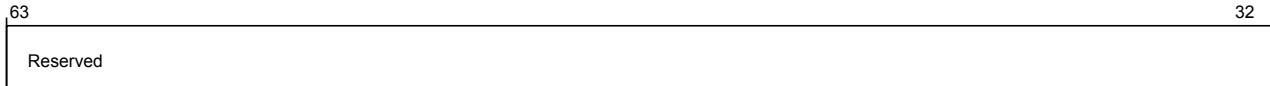


Figure 3-211 por_mtu_por_mtu_tc_flush_sr (high)

The following table shows the por_mtu_tc_flush_sr higher register bit assignments.

Table 3-231 por_mtu_por_mtu_tc_flush_sr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

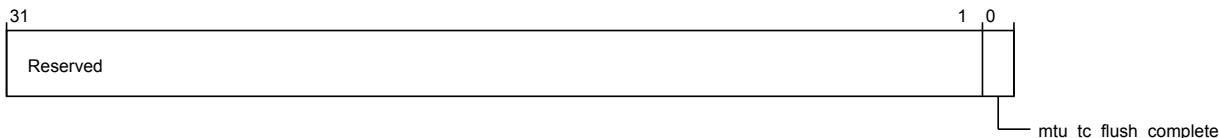


Figure 3-212 por_mtu_por_mtu_tc_flush_sr (low)

The following table shows the por_mtu_tc_flush_sr lower register bit assignments.

Table 3-232 por_mtu_por_mtu_tc_flush_sr (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	mtu_tc_flush_complete	Tag Cache Flush Complete	RO	1'b0

por_mtu_tag_addr_ctl

PA to DA address conversion control

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA40

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

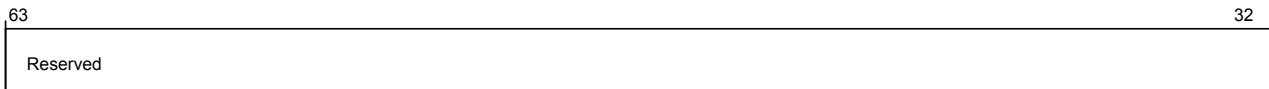


Figure 3-213 por_mtu_por_mtu_tag_addr_ctl (high)

The following table shows the por_mtu_tag_addr_ctl higher register bit assignments.

Table 3-233 por_mtu_por_mtu_tag_addr_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

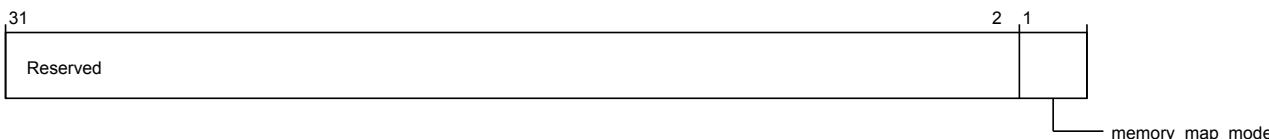


Figure 3-214 por_mtu_por_mtu_tag_addr_ctl (low)

The following table shows the por_mtu_tag_addr_ctl lower register bit assignments.

Table 3-234 por_mtu_por_mtu_tag_addr_ctl (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1:0	memory_map_mode	Memory map mode used for translating data physical address to data DRAM address 2'b00 : Pass-through 2'b01 : PDD 2'b10 : Infra 2'b11 : Infra with 2 sockets	RW	2'b0

por_mtu_tag_addr_base

Physical address of tag base

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA48

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

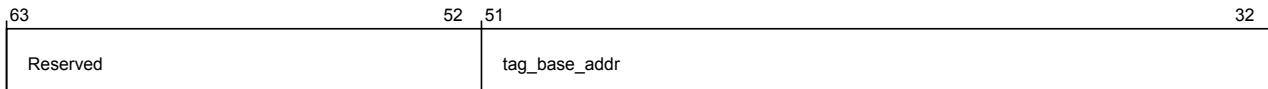


Figure 3-215 por_mtu_por_mtu_tag_addr_base (high)

The following table shows the por_mtu_tag_addr_base higher register bit assignments.

Table 3-235 por_mtu_por_mtu_tag_addr_base (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	tag_base_addr	52-bit Physical address for tag base	RW	52'b0

The following image shows the lower register bit assignments.

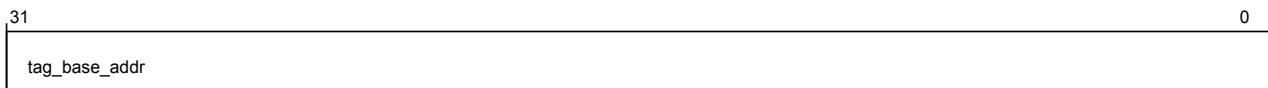


Figure 3-216 por_mtu_por_mtu_tag_addr_base (low)

The following table shows the por_mtu_tag_addr_base lower register bit assignments.

Table 3-236 por_mtu_por_mtu_tag_addr_base (low)

Bits	Field name	Description	Type	Reset
31:0	tag_base_addr	52-bit Physical address for tag base	RW	52'b0

por_mtu_tag_addr_shutter_0-2

This register repeats 2 times. It parametrized by the index from 0 to 2. shutter value to generate DRAM address from physical address

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA50 + #{8*[0, 1, 2]}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

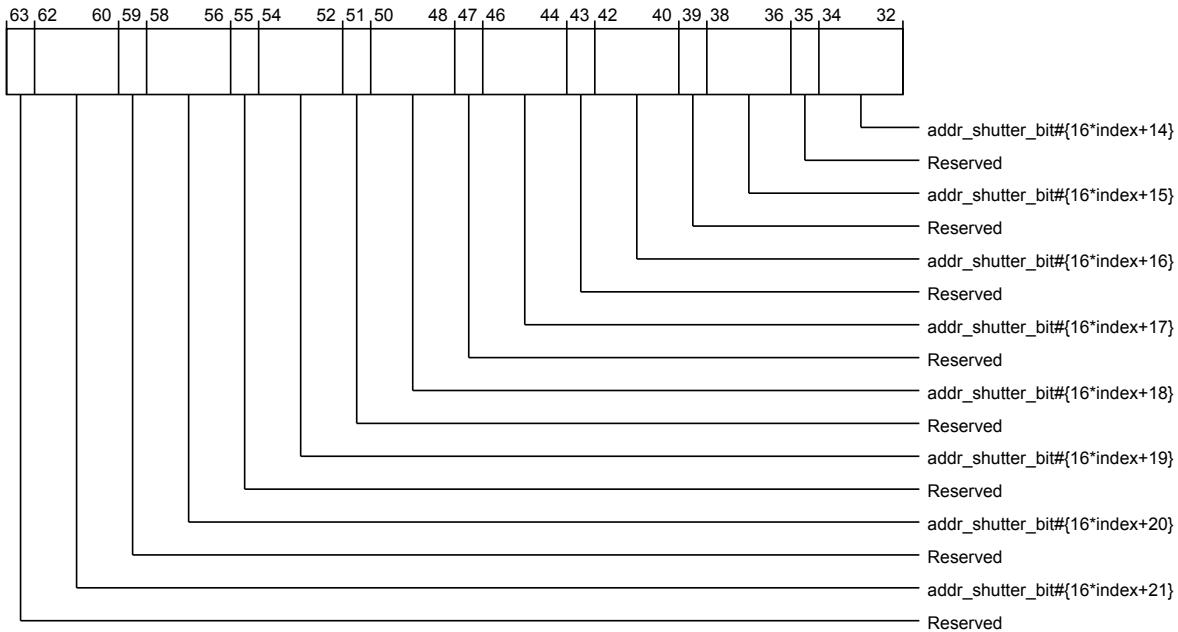


Figure 3-217 por_mtu_por_mtu_tag_addr_shutter_0-2 (high)

The following table shows the por_mtu_tag_addr_shutter_0-2 higher register bit assignments.

Table 3-237 por_mtu_por_mtu_tag_addr_shutter_0-2 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:60	addr_shutter_bit#{16*index+21}	Program to specify how shuttered address bit #{16*index+21} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
59	Reserved	Reserved	RO	-

Table 3-237 por_mtu_por_mtu_tag_addr_shutter_0-2 (high) (continued)

Bits	Field name	Description	Type	Reset
58:56	addr_shutter_bit#{16*index+20}	Program to specify how shuttered address bit #{16*index+20} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
55	Reserved	Reserved	RO	-
54:52	addr_shutter_bit#{16*index+19}	Program to specify how shuttered address bit #{16*index+19} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
51	Reserved	Reserved	RO	-
50:48	addr_shutter_bit#{16*index+18}	Program to specify how shuttered address bit #{16*index+18} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
47	Reserved	Reserved	RO	-

Table 3-237 por_mtu_por_mtu_tag_addr_shutter_0-2 (high) (continued)

Bits	Field name	Description	Type	Reset
46:44	addr_shutter_bit#{16*index+17}	Program to specify how shuttered address bit #{16*index+17} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
43	Reserved	Reserved	RO	-
42:40	addr_shutter_bit#{16*index+16}	Program to specify how shuttered address bit #{16*index+16} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
39	Reserved	Reserved	RO	-
38:36	addr_shutter_bit#{16*index+15}	Program to specify how shuttered address bit #{16*index+15} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0

Table 3-237 por_mtu_por_mtu_tag_addr_shutter_0-2 (high) (continued)

Bits	Field name	Description	Type	Reset
35	Reserved	Reserved	RO	-
34:32	addr_shutter_bit#{16*index+14}	Program to specify how shuttered address bit # $\{16*index+14\}$ should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0

The following image shows the lower register bit assignments.

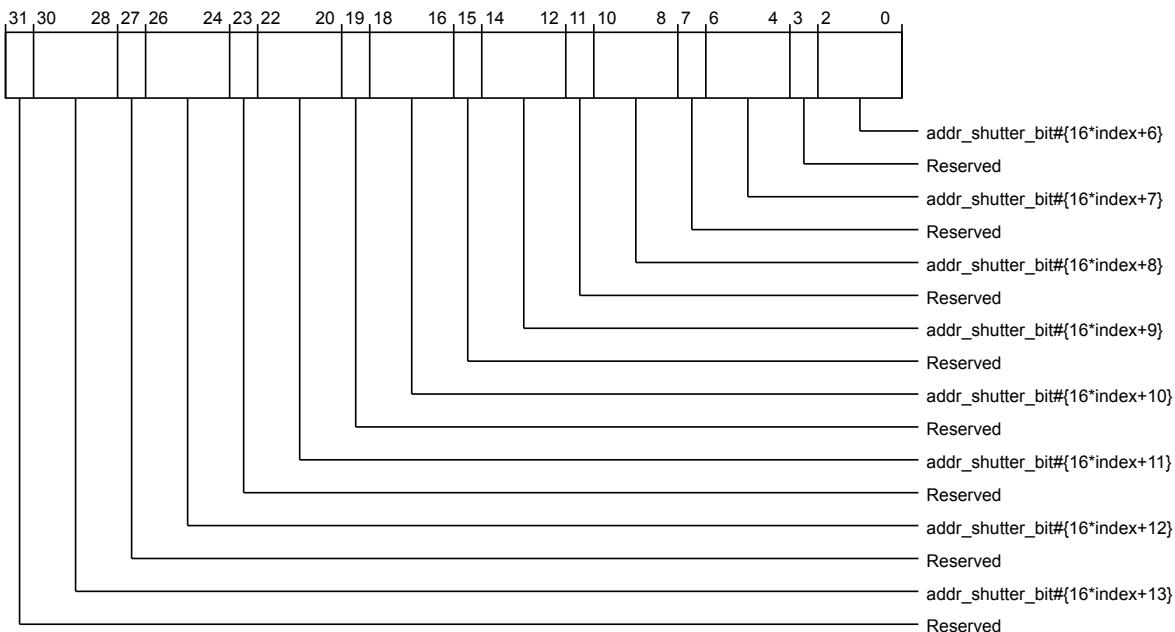


Figure 3-218 por_mtu_por_mtu_tag_addr_shutter_0-2 (low)

The following table shows the por_mtu_tag_addr_shutter_0-2 lower register bit assignments.

Table 3-238 por_mtu_por_mtu_tag_addr_shutter_0-2 (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	addr_shutter_bit#{16*index+13}	Program to specify how shuttered address bit #{16*index+13} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
27	Reserved	Reserved	RO	-
26:24	addr_shutter_bit#{16*index+12}	Program to specify how shuttered address bit #{16*index+12} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
23	Reserved	Reserved	RO	-
22:20	addr_shutter_bit#{16*index+11}	Program to specify how shuttered address bit #{16*index+11} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
19	Reserved	Reserved	RO	-

Table 3-238 por_mtu_por_mtu_tag_addr_shutter_0-2 (low) (continued)

Bits	Field name	Description	Type	Reset
18:16	addr_shutter_bit#{16*index+10}	Program to specify how shuttered address bit #{16*index+10} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
15	Reserved	Reserved	RO	-
14:12	addr_shutter_bit#{16*index+9}	Program to specify how shuttered address bit #{16*index+9} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
11	Reserved	Reserved	RO	-
10:8	addr_shutter_bit#{16*index+8}	Program to specify how shuttered address bit #{16*index+8} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
7	Reserved	Reserved	RO	-

Table 3-238 por_mtu_por_mtu_tag_addr_shutter_0-2 (low) (continued)

Bits	Field name	Description	Type	Reset
6:4	addr_shutter_bit#{16*index+7}	Program to specify how shuttered address bit #{16*index+7} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0
3	Reserved	Reserved	RO	-
2:0	addr_shutter_bit#{16*index+6}	Program to specify how shuttered address bit #{16*index+6} should be driven from post-translation address 3'b000 : pass-through 3'b001 : shift_1 3'b010 : shift_2 3'b011 : shift_3 3'b100 : shift_4 3'b101 : shift_5 3'b110 : shift_6 3'b111 : shift_7	RW	3'b0

por_mtu_errfr

Functions as the error feature register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3000
Register reset	64'b1001010100001
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mtu_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

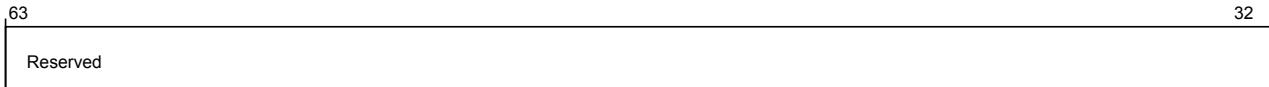


Figure 3-219 por_mtu_por_mtu_errfr (high)

The following table shows the por_mtu_errfr higher register bit assignments.

Table 3-239 por_mtu_por_mtu_errfr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

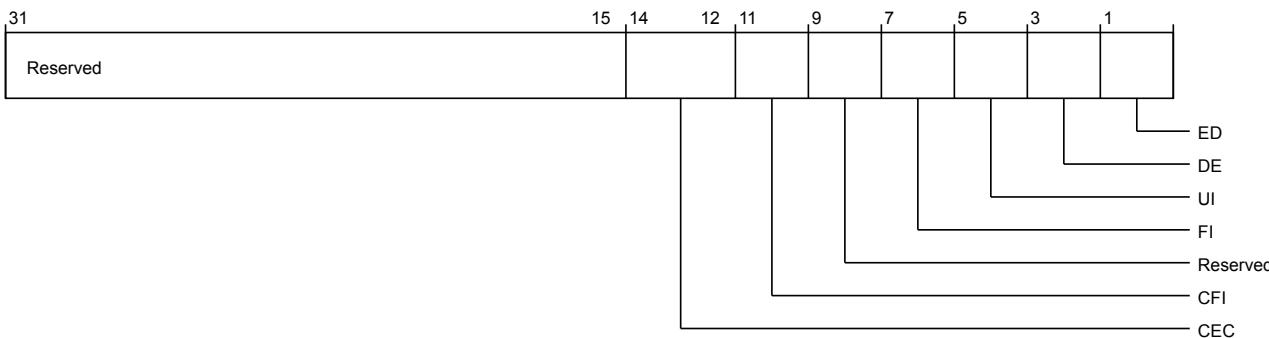


Figure 3-220 por_mtu_por_mtu_errfr (low)

The following table shows the por_mtu_errfr lower register bit assignments.

Table 3-240 por_mtu_por_mtu_errfr (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_mtu_errmisc[39:32] 3'b100: Implements 16-bit error counter in por_mtu_errmisc[47:32]	RO	3'b100
11:10	CFI	Corrected error interrupt	RO	2'b10
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10

Table 3-240 por_mtu_por_mtu_errfr (low) (continued)

Bits	Field name	Description	Type	Reset
3:2	DE	Deferred errors	RO	2'b00
1:0	ED	Error detection 2'b00: Feature not supported 2'b01: Feature always enabled 2'b10: Feature is controllable 2'b11: Reserved	RO	2'b01

por_mtu_errctlr

Functions as the error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3008
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group	por_mtu_secure_register_groups_override.ras_secure_access_override_override

The following image shows the higher register bit assignments.

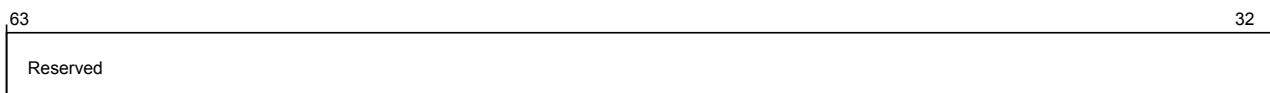


Figure 3-221 por_mtu_por_mtu_errctlr (high)

The following table shows the por_mtu_errctlr higher register bit assignments.

Table 3-241 por_mtu_por_mtu_errctlr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

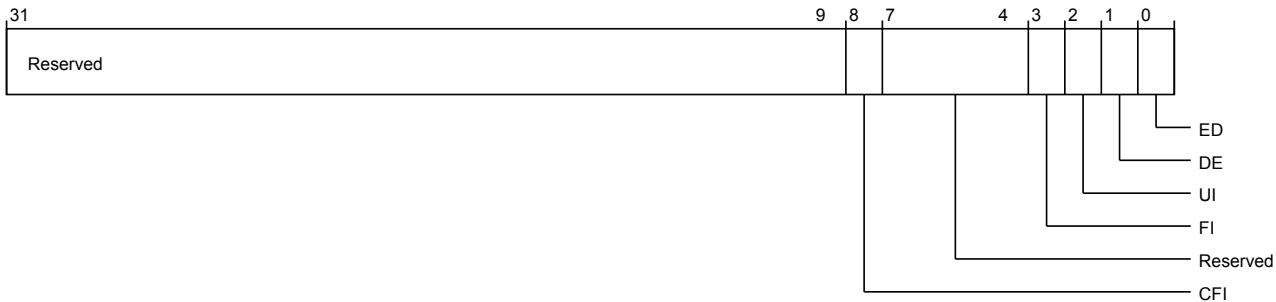


Figure 3-222 por_mtu_por_mtu_errctlr (low)

The following table shows the por_mtu_errctlr lower register bit assignments.

Table 3-242 por_mtu_por_mtu_errctlr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_mtu_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_mtu_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_mtu_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_mtu_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_mtu_errfr.ED	RW	1'b0

por_mtu_errstatus

Functions as the error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3010
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mtu_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

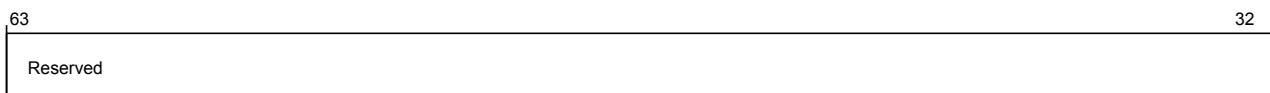


Figure 3-223 por_mtu_por_mtu_errstatus (high)

The following table shows the por_mtu_errstatus higher register bit assignments.

Table 3-243 por_mtu_por_mtu_errstatus (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-224 por_mtu_por_mtu_errstatus (low)

The following table shows the por_mtu_errstatus lower register bit assignments.

Table 3-244 por_mtu_por_mtu_errstatus (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_mtu_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0

Table 3-244 por_mtu_por_mtu_errstatus (low) (continued)

Bits	Field name	Description	Type	Reset
26	MV	por_mtu_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_mtu_erraddr

Contains the error record address.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3018

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group por_mtu_secure_register_groups_override.ras_secure_access_override
override

The following image shows the higher register bit assignments.

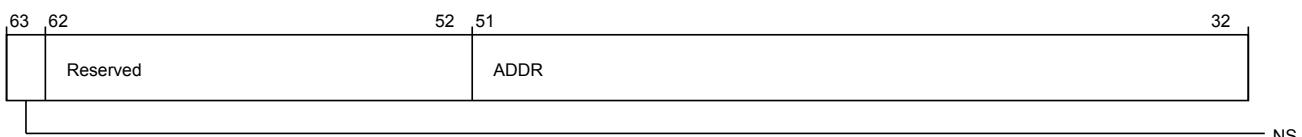


Figure 3-225 por_mtu_por_mtu_erraddr (high)

The following table shows the por_mtu_erraddr higher register bit assignments.

Table 3-245 por_mtu_por_mtu_erraddr (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_mtu_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address for Data Request (Data PA). Same PA received from HN. Note: For Errors on TC evictions, this represents address that caused eviction. Note: Address is undefined if errmisc.OPTYPE is Other. For most part, these transactions are set/way based and not PA based.	RW	52'b0

The following image shows the lower register bit assignments.

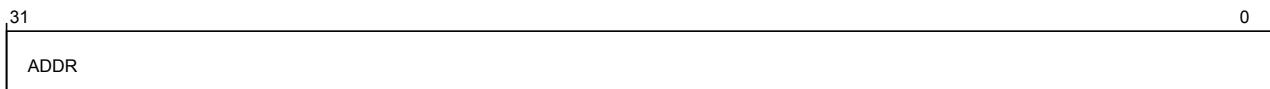


Figure 3-226 por_mtu_por_mtu_erraddr (low)

The following table shows the por_mtu_erraddr lower register bit assignments.

Table 3-246 por_mtu_por_mtu_erraddr (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address for Data Request (Data PA). Same PA received from HN. Note: For Errors on TC evictions, this represents address that caused eviction. Note: Address is undefined if errmisc.OPTYPE is Other. For most part, these transactions are set/way based and not PA based.	RW	52'b0

por_mtu_errmisc

Functions as the miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3020
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mtu_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

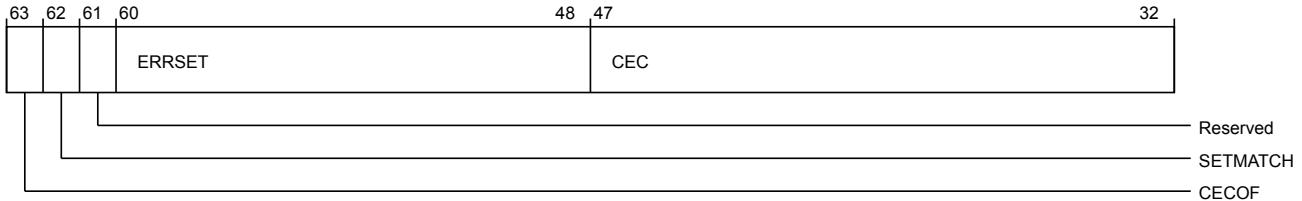


Figure 3-227 por_mtu_por_mtu_errmisc (high)

The following table shows the por_mtu_errmisc higher register bit assignments.

Table 3-247 por_mtu_por_mtu_errmisc (high)

Bits	Field name	Description	Type	Reset
63	CECOF	Corrected error counter overflow	RW	1'b0
62	SETMATCH	Set address match	RW	1'b0
61	Reserved	Reserved	RO	-
60:48	ERRSET	TC set address for ECC Single bit error	RW	13'b0
47:32	CEC	Corrected ECC error count	RW	16'b0

The following image shows the lower register bit assignments.

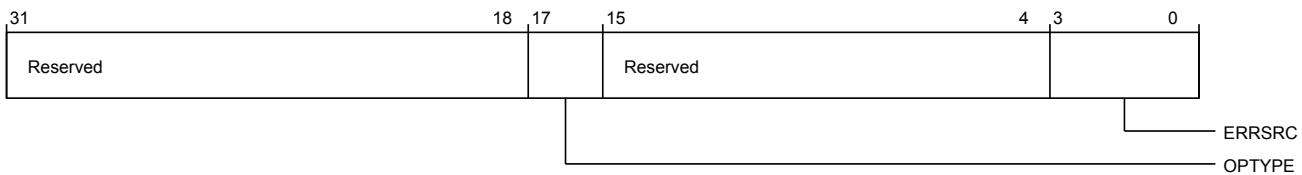


Figure 3-228 por_mtu por_mtu_errmisc (low)

The following table shows the port mtu errmisc lower register bit assignments.

Table 3-248 por mtu por mtu errmisc (low)

Bits	Field name	Description	Type	Reset
31:18	Reserved	Reserved	RO	-
17:16	OPTYPE	Error opcode type 2'b00: Read Type (RD_NO_SNP, PrefetchTgt) 2'b01: Write (WR_NO_SNP) 2'b10: CMO, WR+CMO 2'b11: Other	RW	2'b00

Table 3-248 por_mtu_por_mtu_errmisc (low) (continued)

Bits	Field name	Description	Type	Reset
15:4	Reserved	Reserved	RO	-
3:0	ERRSRC	Error source 4'b0001: Data single-bit ECC 4'b0010: Data double-bit ECC 4'b0011: Single-bit ECC overflow 4'b0101: Control single-bit ECC 4'b0110: Control double-bit ECC 4'b1000: AXI AR Slave Error 4'b1001: AXI AR Decode Error 4'b1010: AXI AR Poison Error 4'b1011: AXI AR Datachk Error 4'b1100: AXI W Slave Error 4'b1101: AXI W Decode Error 4'b1110: PA out of range Error	RW	4'b0000

por_mtu_errfr_NS

Functions as the non-secure error feature register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3100

Register reset 64'b1001010100001

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

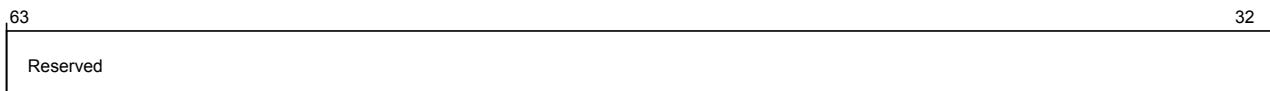


Figure 3-229 por_mtu_por_mtu_errfr_ns (high)

The following table shows the por_mtu_errfr_NS higher register bit assignments.

Table 3-249 por_mtu_por_mtu_errfr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

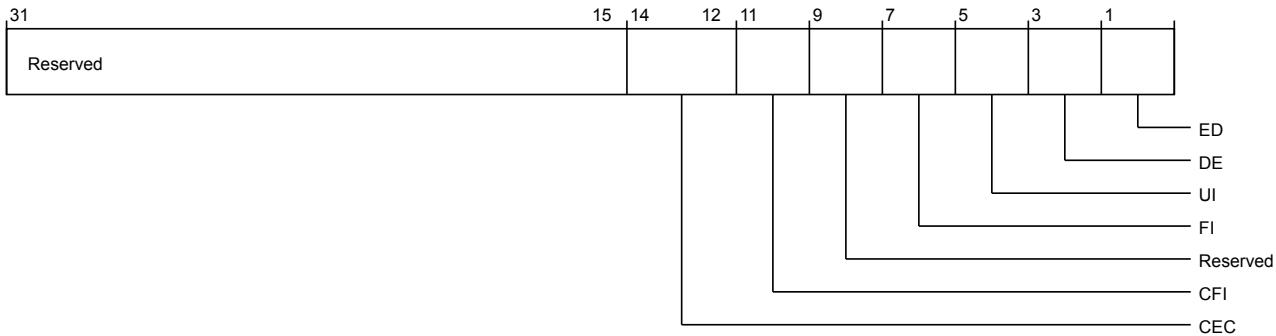


Figure 3-230 por_mtu_por_mtu_errfr_ns (low)

The following table shows the por_mtu_errfr_NS lower register bit assignments.

Table 3-250 por_mtu_por_mtu_errfr_ns (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_mtu_errmisc_NS[39:32] 3'b100: Implements 16-bit error counter in por_mtu_errmisc_NS[47:32]	RO	3'b100
11:10	CFI	Corrected error interrupt	RO	2'b10
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors	RO	2'b00
1:0	ED	Error detection 2'b00: Feature not supported 2'b01: Feature always enabled 2'b10: Feature is controllable 2'b11: Reserved	RO	2'b01

por_mtu_errctlr_NS

Functions as the non-secure error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3108

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

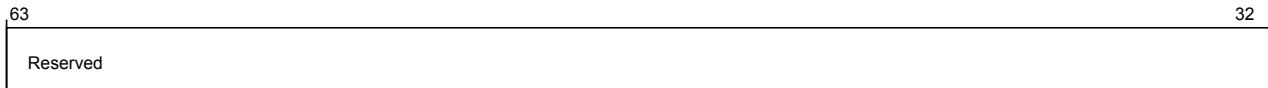


Figure 3-231 por_mtu_por_mtu_errctlr_ns (high)

The following table shows the por_mtu_errctlr_NS higher register bit assignments.

Table 3-251 por_mtu_por_mtu_errctlr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

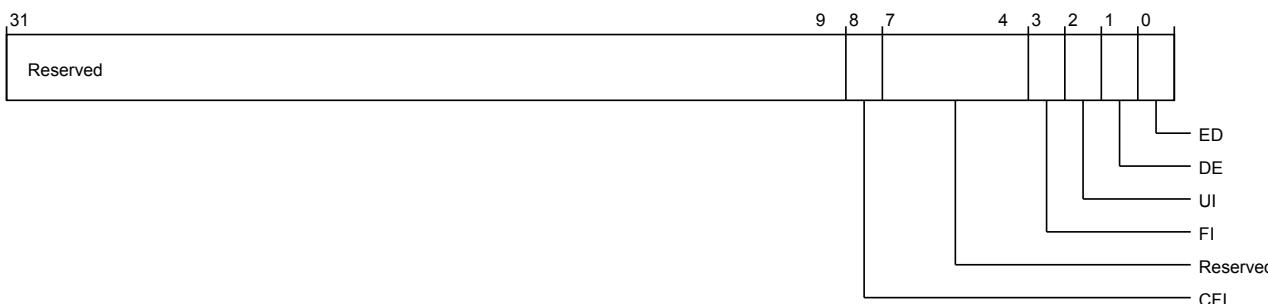


Figure 3-232 por_mtu_por_mtu_errctlr_ns (low)

The following table shows the por_mtu_errctlr_NS lower register bit assignments.

Table 3-252 por_mtu_por_mtu_errctlr_ns (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_mtu_errfr_NS.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_mtu_errfr_NS.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_mtu_errfr_NS.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_mtu_errfr_NS.DE	RW	1'b0
0	ED	Enables error detection as specified in por_mtu_errfr_NS.ED	RW	1'b0

por_mtu_errstatus_NS

Functions as the non-secure error status register.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3110
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

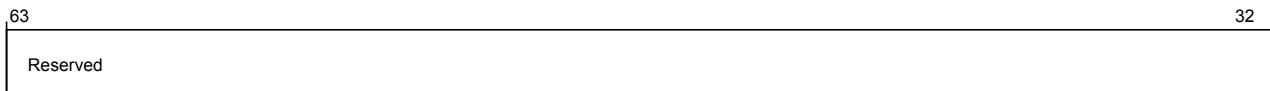


Figure 3-233 por_mtu_por_mtu_errstatus_ns (high)

The following table shows the por_mtu_errstatus_NS higher register bit assignments.

Table 3-253 por_mtu_por_mtu_errstatus_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

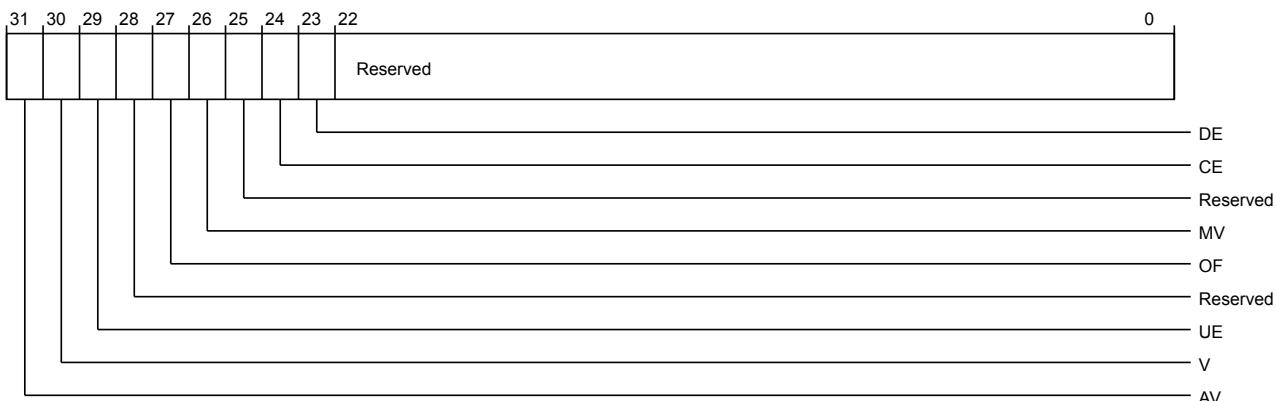


Figure 3-234 por_mtu_por_mtu_errstatus_ns (low)

The following table shows the port mtu errstatus NS lower register bit assignments.

Table 3-254 por_mtu_por_mtu_errstatus_ns (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_mtu_erraddr_NS contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_mtu_errmisc_NS valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_mtu_erraddr_NS

Contains the non-secure error record address.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3118
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

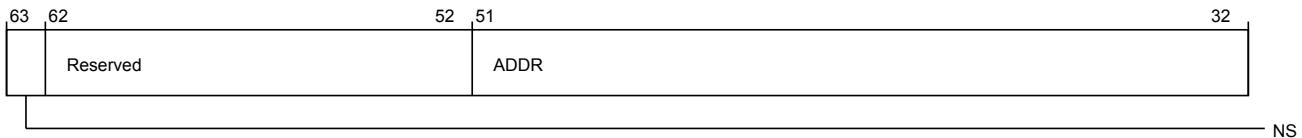


Figure 3-235 por_mtu_por_mtu_erraddr_ns (high)

The following table shows the por_mtu_erraddr_NS higher register bit assignments.

Table 3-255 por_mtu_por_mtu_erraddr_ns (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_mtu_erraddr_NS.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address for Data Request (Data PA). Same PA received from HN. Note: For Errors on TC evictions, this represents address that caused eviction. Note: Address is undefined if ermisc.OPTYPE is Other. For most part, these transactions are set/way based and not PA based.	RW	52'b0

The following image shows the lower register bit assignments.

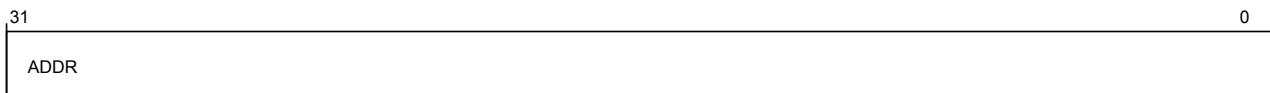


Figure 3-236 por_mtu_por_mtu_erraddr_ns (low)

The following table shows the por_mtu_erraddr_NS lower register bit assignments.

Table 3-256 por_mtu_por_mtu_erraddr_ns (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address for Data Request (Data PA). Same PA received from HN. Note: For Errors on TC evictions, this represents address that caused eviction. Note: Address is undefined if errmisc.OPTYPE is Other. For most part, these transactions are set/way based and not PA based.	RW	52'b0

por_mtu_errmisc_NS

Functions as the non-secure miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3120

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

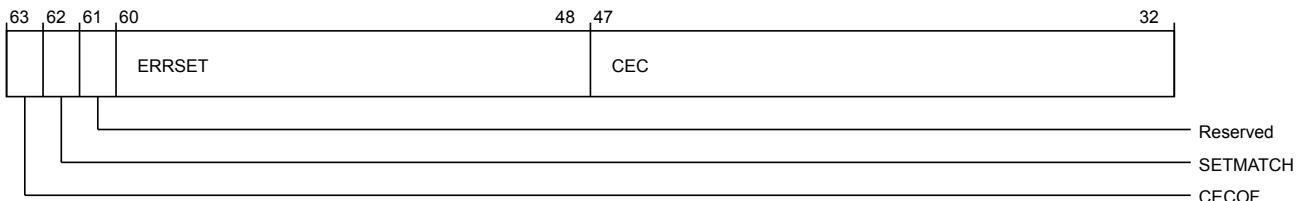


Figure 3-237 por_mtu_por_mtu_errmisc_ns (high)

The following table shows the por_mtu_errmisc_NS higher register bit assignments.

Table 3-257 por_mtu_por_mtu_errmisc_ns (high)

Bits	Field name	Description	Type	Reset
63	CECOF	Corrected error counter overflow	RW	1'b0
62	SETMATCH	Set address match	RW	1'b0
61	Reserved	Reserved	RO	-
60:48	ERRSET	TC set address for ECC Single bit error	RW	13'b0
47:32	CEC	Corrected ECC error count	RW	16'b0

The following image shows the lower register bit assignments.

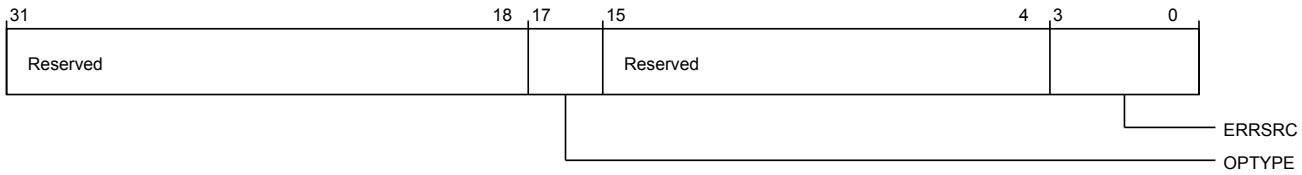


Figure 3-238 por_mtu_por_mtu_errmisc_ns (low)

The following table shows the por_mtu_errmisc_NS lower register bit assignments.

Table 3-258 por_mtu_por_mtu_errmisc_ns (low)

Bits	Field name	Description	Type	Reset
31:18	Reserved	Reserved	RO	-
17:16	OPTYPE	Error opcode type 2'b00: Read Type (RD_NO_SNP, PrefetchTgt) 2'b01: Write (WR_NO_SNP) 2'b10: CMO, WR+CMO 2'b11: Other op types	RW	2'b00
15:4	Reserved	Reserved	RO	-
3:0	ERRSRC	Error source 4'b0001: Data single-bit ECC 4'b0010: Data double-bit ECC 4'b0011: Single-bit ECC overflow 4'b0101: Control single-bit ECC 4'b0110: Control double-bit ECC 4'b1000: AXI AR Slave Error 4'b1001: AXI AR Decode Error 4'b1010: AXI AR Poison Error 4'b1011: AXI AR Datachk Error 4'b1100: AXI W Slave Error 4'b1101: AXI W Decode Error 4'b1110: PA out of range Error	RW	4'b0000

por_mtu_err_inj

Enables error injection and setup. When enabled for a given PA and NS bit, MTU returns an error interrupt which emulates a TC double-bit data ECC error. This feature enables software to test the error handler. The error is reported for cacheable read access with Tag Op Transfer for which TC hit. No error is reported for cacheable read access for which TC miss.

Its characteristics are:

Type RW

Register width (Bits)	64
Address offset	16'h3030
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

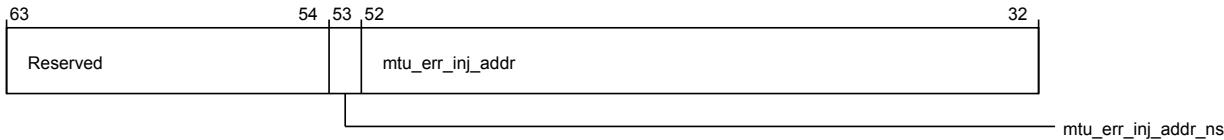


Figure 3-239 por_mtu_por_mtu_err_inj (high)

The following table shows the por_mtu_err_inj higher register bit assignments.

Table 3-259 por_mtu_por_mtu_err_inj (high)

Bits	Field name	Description	Type	Reset
63:54	Reserved	Reserved	RO	-
53	mtu_err_inj_addr_ns	Address NS used to match for error injection	RW	1'b0
52:32	mtu_err_inj_addr	Physical Address used to match for error injection	RW	52'b0

The following image shows the lower register bit assignments.

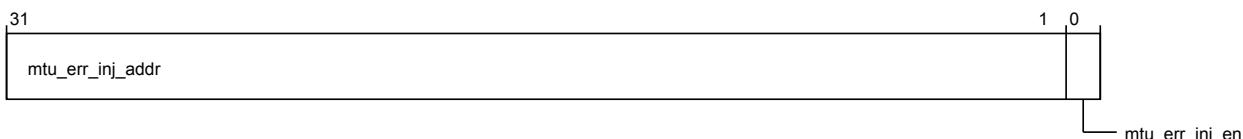


Figure 3-240 por_mtu_por_mtu_err_inj (low)

The following table shows the por_mtu_err_inj lower register bit assignments.

Table 3-260 por_mtu_por_mtu_err_inj (low)

Bits	Field name	Description	Type	Reset
31:1	mtu_err_inj_addr	Physical Address used to match for error injection	RW	52'b0
0	mtu_err_inj_en	Enables error injection and report	RW	1'b0

por_mtu_cfg_tc_dbgrd

Controls access modes for TC data and TC Control debug read.

Its characteristics are:

Type WO

Register width (Bits) 64

Address offset	16'hB80
Register reset	64'b10000000000000000000000000000000
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mtu_secure_register_groups_override.tc_dbgrd

The following image shows the higher register bit assignments.

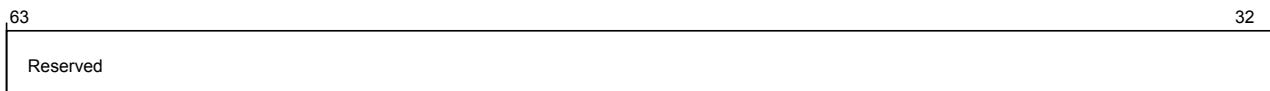


Figure 3-241 por_mtu_por_mtu_cfg_tc_dbgrd (high)

The following table shows the por_mtu_cfg_tc_dbgrd higher register bit assignments.

Table 3-261 por_mtu_por_mtu_cfg_tc_dbgrd (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

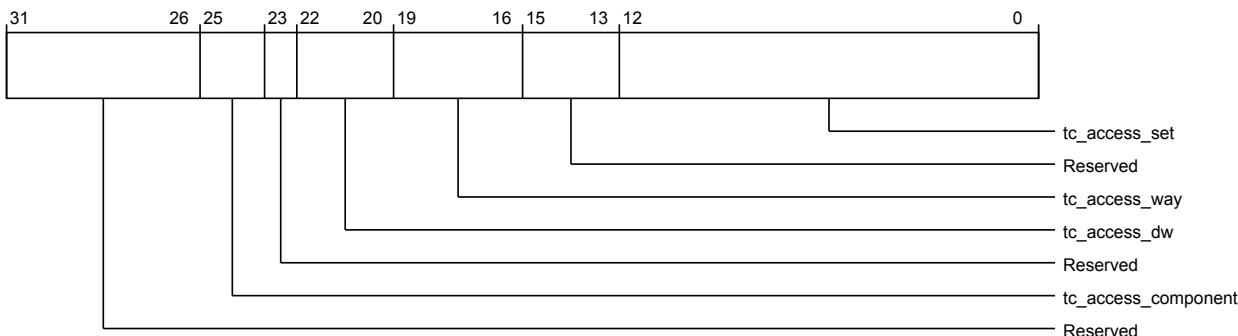


Figure 3-242 por_mtu_por_mtu_cfg_tc_dbgrd (low)

The following table shows the por_mtu_cfg_tc_dbgrd lower register bit assignments.

Table 3-262 por_mtu_por_mtu_cfg_tc_dbgrd (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	tc_access_component	Specifies TC Data/Control array debug read 2'b01: TC data read 2'b10: TC control read	WO	2'b10
23	Reserved	Reserved	RO	-
22:20	tc_access_dw	64-bit chunk address for TC data debug read access	WO	3'h0
19:16	tc_access_way	Way address for TC debug read access	WO	4'h0

Table 3-262 por_mtu_por_mtu_cfg_tc_dbgrd (low) (continued)

Bits	Field name	Description	Type	Reset
15:13	Reserved	Reserved	RO	-
12:0	tc_access_set	Set address for TC debug read access	WO	13'h0

por_mtu_tc_cache_access_tc_ctl

Contains TC Control debug read data bits

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hB88

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_mtu_secure_register_groups_override.tc_dbgrd

The following image shows the higher register bit assignments.

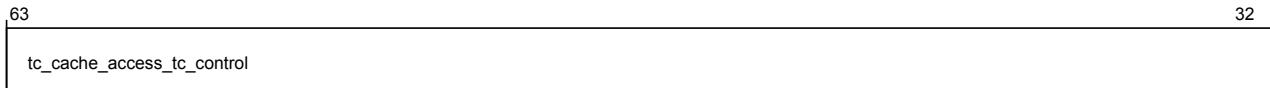


Figure 3-243 por_mtu_por_mtu_tc_cache_access_tc_ctl (high)

The following table shows the por_mtu_tc_cache_access_tc_ctl higher register bit assignments.

Table 3-263 por_mtu_por_mtu_tc_cache_access_tc_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	tc_cache_access_tc_control	TC Control debug read data	RO	64'h0

The following image shows the lower register bit assignments.

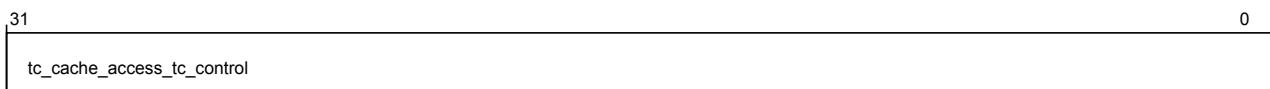


Figure 3-244 por_mtu_por_mtu_tc_cache_access_tc_ctl (low)

The following table shows the por_mtu_tc_cache_access_tc_ctl lower register bit assignments.

Table 3-264 por_mtu_por_mtu_tc_cache_access_tc_ctl (low)

Bits	Field name	Description	Type	Reset
31:0	tc_cache_access_tc_control	TC Control debug read data	RO	64'h0

por_mtu_tc_cache_access_tc_data

Contains TC data RAM debug read data.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'hB98
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mtu_secure_register_groups_override.tc_dbgrd

The following image shows the higher register bit assignments.

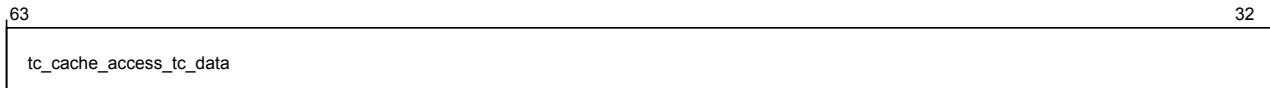


Figure 3-245 por_mtu_por_mtu_tc_cache_access_tc_data (high)

The following table shows the por_mtu_tc_cache_access_tc_data higher register bit assignments.

Table 3-265 por_mtu_por_mtu_tc_cache_access_tc_data (high)

Bits	Field name	Description	Type	Reset
63:32	tc_cache_access_tc_data	TC data RAM debug read data	RO	64'h0

The following image shows the lower register bit assignments.



Figure 3-246 por_mtu_por_mtu_tc_cache_access_tc_data (low)

The following table shows the por_mtu_tc_cache_access_tc_data lower register bit assignments.

Table 3-266 por_mtu_por_mtu_tc_cache_access_tc_data (low)

Bits	Field name	Description	Type	Reset
31:0	tc_cache_access_tc_data	TC data RAM debug read data	RO	64'h0

por_mtu_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

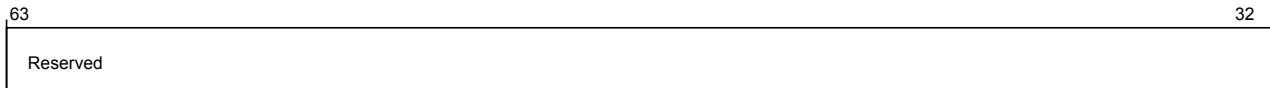


Figure 3-247 por_mtu_por_mtu_pmu_event_sel (high)

The following table shows the por_mtu_por_mtu_pmu_event_sel higher register bit assignments.

Table 3-267 por_mtu_por_mtu_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

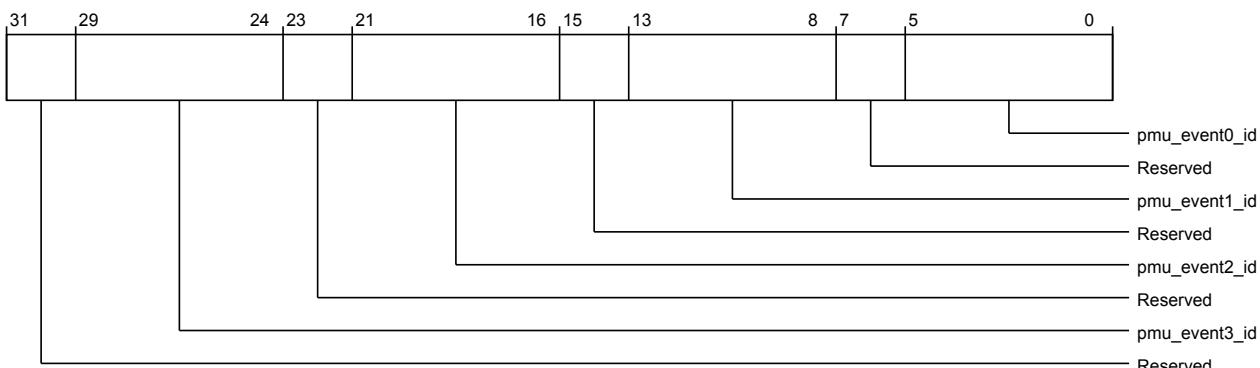


Figure 3-248 por_mtu_por_mtu_pmu_event_sel (low)

The following table shows the por_mtu_pmu_event_sel lower register bit assignments.

Table 3-268 por_mtu_por_mtu_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	MTU PMU Event 3 select; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	MTU PMU Event 2 select; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	MTU PMU Event 1 select; see pmu_event0_id for encodings	RW	6'b0

Table 3-268 por_mtu_por_mtu_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-
5:0	pmu_event0_id	<p>MTU PMU Event 0 select</p> <p>6'h00: No event</p> <p>6'h01: PMU_MTU_TC_LOOKUP_EVENT; Count total cache lookup requests.</p> <p>6'h02: PMU_MTU_TC_FILL_EVENT; Count total number of tag cache allocation (Dirty or Clean) requests.</p> <p>6'h03: PMU_MTU_TC_MISS_EVENT; Count total cache miss responses.</p> <p>6'h04: PMU_MTU_TDB_FORWARD_EVENT; Count total number of requests that got TDB forwarded data.</p> <p>6'h05: PMU_MTU_TCQ_HAZARD_EVENT; Count number of incoming requests hazarding against pending TCQ requests.</p> <p>6'h06: PMU_MTU_TCQ_RD_ALLOC_EVENT; Count number of read requests allocated in TCQ. This includes Read and Write_Match.</p> <p>6'h07: PMU_MTU_TCQ_WR_ALLOC_EVENT; Count number of write requests allocated in TCQ. This includes Write_Update.</p> <p>6'h08: PMU_MTU_TCQ_CMO_ALLOC_EVENT; Count number of CMO requests allocated in TCQ. This includes CMOs and wr+CMO.</p> <p>6'h09: PMU_MTU_AXI_RD_REQ_EVENT; Count number of read requests sent out on AXI.</p> <p>6'h0A: PMU_MTU_AXI_WR_REQ_EVENT; Count number of write requests sent out on AXI.</p> <p>6'h0B: PMU_MTU_TCQ_OCCUPANCY_CNT_OVERFLOW_EVENT; TCQ tracker occupancy count overflow.</p> <p>6'h0C: PMU_MTU_TDB_OCCUPANCY_CNT_OVERFLOW_EVENT; TDB occupancy count overflow.</p> <p>NOTE: All other encodings are reserved.</p>	RW	6'b0

3.3.6 CXLA register descriptions

This section lists the CXLA registers.

por_cxla_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	48 47	32
Reserved	logical_id	

Figure 3-249 por_cxla_por_cxla_node_info (high)

The following table shows the por_cxla_node_info higher register bit assignments.

Table 3-269 por_cxla_por_cxla_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

31	16 15	0
node_id	node_type	

Figure 3-250 por_cxla_por_cxla_node_info (low)

The following table shows the por_cxla_node_info lower register bit assignments.

Table 3-270 por_cxla_por_cxla_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0102

por_cxla_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-251 por_cxla_por_cxla_child_info (high)

The following table shows the por_cxla_child_info higher register bit assignments.

Table 3-271 por_cxla_por_cxla_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

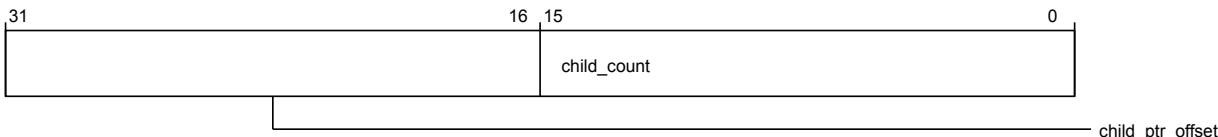


Figure 3-252 por_cxla_por_cxla_child_info (low)

The following table shows the por_cxla_child_info lower register bit assignments.

Table 3-272 por_cxla_por_cxla_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_cxla_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-253 por_cxla_por_cxla_secure_register_groups_override (high)

The following table shows the por_cxla_secure_register_groups_override higher register bit assignments.

Table 3-273 por_cxla_por_cxla_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

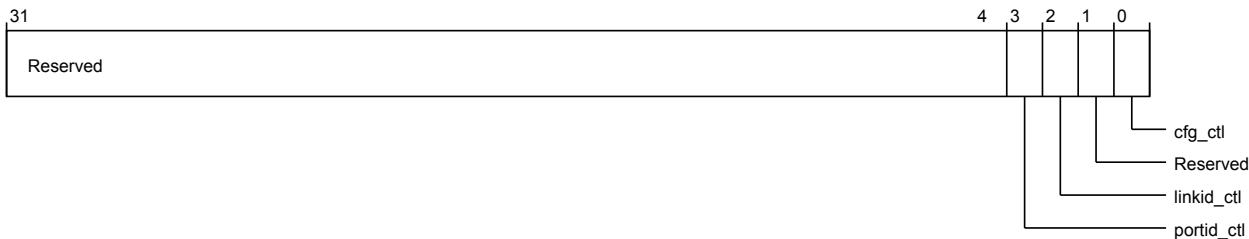


Figure 3-254 por_cxla_por_cxla_secure_register_groups_override (low)

The following table shows the por_cxla_secure_register_groups_override lower register bit assignments.

Table 3-274 por_cxla_por_cxla_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	portid_ctl	Allows non-secure access to secure LA Port ID registers	RW	1'b0
2	linkid_ctl	Allows non-secure access to secure LA Link ID registers	RW	1'b0
1	Reserved	Reserved	RO	-
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_cxla_unit_info

Provides component identification information for CXLA.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

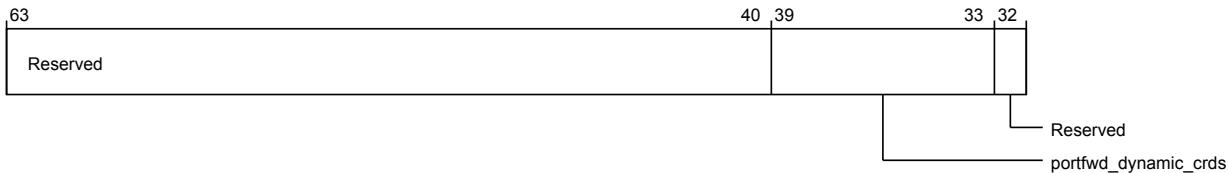


Figure 3-255 por_cxla_por_cxla_unit_info (high)

The following table shows the por_cxla_unit_info higher register bit assignments.

Table 3-275 por_cxla_por_cxla_unit_info (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:33	portfwd_dynamic_crds	Number of dynamic credits granted by this CXLA port for port forwarded traffic	RO	Configuration dependent
32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

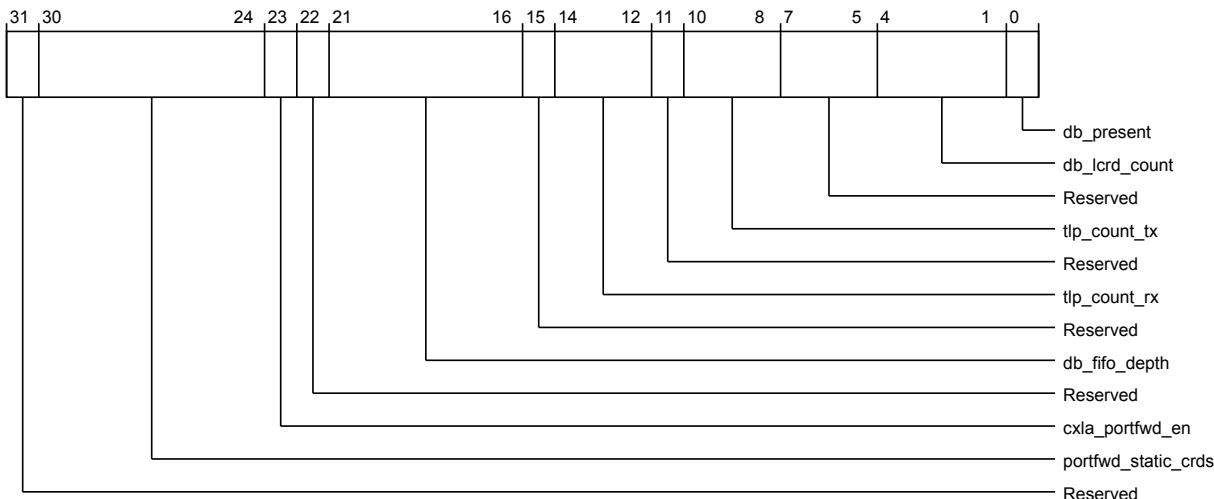


Figure 3-256 por_cxla_por_cxla_unit_info (low)

The following table shows the por_cxla_unit_info lower register bit assignments.

Table 3-276 por_cxla_por_cxla_unit_info (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:24	portfwd_static_crds	Number of static credits granted by this CXLA port for port forwarded traffic	RO	Configuration dependent
23	cxla_portfwd_en	Port forwarding is enabled at this CXLA port	RO	Configuration dependent
22	Reserved	Reserved	RO	-

Table 3-276 por_cxla_por_cxla_unit_info (low) (continued)

Bits	Field name	Description	Type	Reset
21:16	db_fifo_depth	FIFO Depth in CXLA Domain Bridges - CXDB, PDB	RO	Configuration dependent
15	Reserved	Reserved	RO	-
14:12	tlp_count_rx	Maximum number of TLPs supported by RX TLP buffer	RO	Configuration dependent
11	Reserved	Reserved	RO	-
10:8	tlp_count_tx	Maximum number of TLPs supported by TX TLP buffer	RO	Configuration dependent
7:5	Reserved	Reserved	RO	-
4:1	db_lcrd_count	Number of flit credits between CXG and CXLA	RO	Configuration dependent
0	db_present	DB present in CXLA	RO	Configuration dependent

por_cxla_cfg_ctl

Functions as the configuration control register for CXLA.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b001001001
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_cxla_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

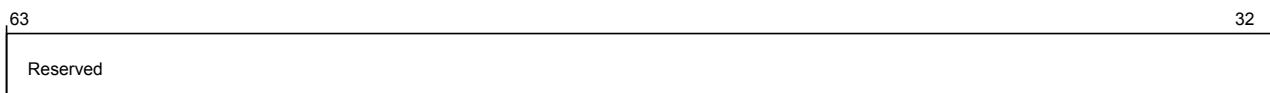


Figure 3-257 por_cxla_por_cxla_cfg_ctl (high)

The following table shows the por_cxla_cfg_ctl higher register bit assignments.

Table 3-277 por_cxla_por_cxla_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

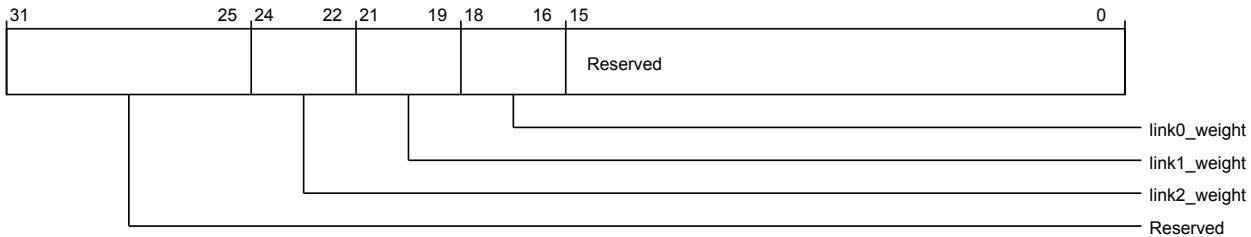


Figure 3-258 por_cxla_por_cxla_cfg_ctl (low)

The following table shows the por_cxla_cfg_ctl lower register bit assignments.

Table 3-278 por_cxla_por_cxla_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24:22	link2_weight	Determines weight of link2 to start forming a TLP in presence of pending messages to other links; applies for message packing	RW	3'b001
21:19	link1_weight	Determines weight of link1 to start forming a TLP in presence of pending messages to other links; applies for message packing	RW	3'b001
18:16	link0_weight	Determines weight of link0 to start forming a TLP in presence of pending messages to other links; applies for message packing	RW	3'b001
15:0	Reserved	Reserved	RO	-

por_cxla_aux_ctl

Functions as the auxiliary control register for CXLA.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	64'b0001001001001001001001001000010
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

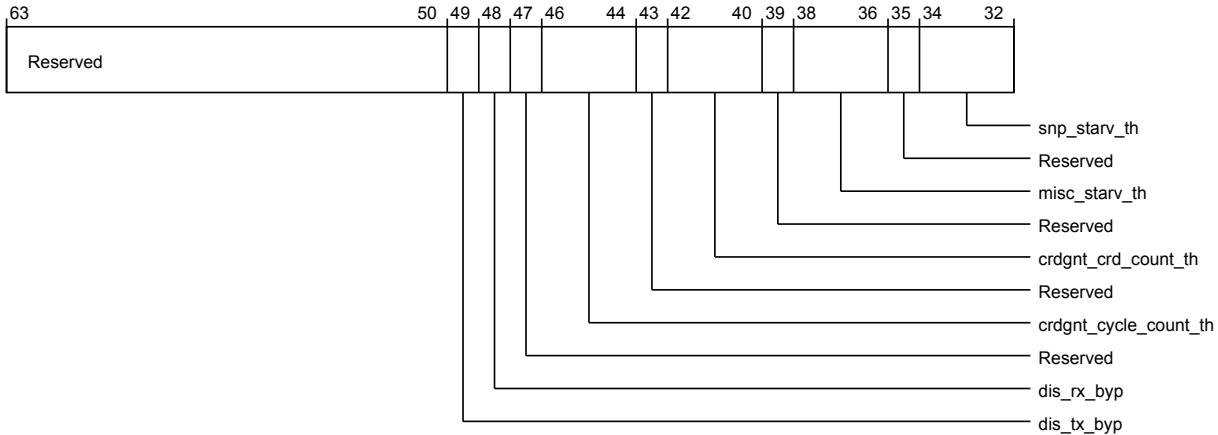


Figure 3-259 por_cxla_por_cxla_aux_ctl (high)

The following table shows the por_cxla_aux_ctl higher register bit assignments.

Table 3-279 por_cxla_por_cxla_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:50	Reserved	Reserved	RO	-
49	dis_tx_byp	When set, disables TX bypass paths	RW	1'b0
48	dis_rx_byp	When set, disables RX bypass paths	RW	1'b0
47	Reserved	Reserved	RO	-
46:44	crdgnt_cycle_count_th	Maximum number of cycles that need to be elapsed since the end of previous TLP to send a credit grant message 3'b000: 32 cycles 3'b001: 64 cycles 3'b010: 128 cycles 3'b011: 256 cycles	RW	3'b010
43	Reserved	Reserved	RO	-
42:40	crdgnt_crd_count_th	Maximum number of credits that need to be accumulated to send a credit grant message 3'b000: 16 credits 3'b001: 32 credits 3'b010: 64 credits 3'b011: 128 credits	RW	3'b010
39	Reserved	Reserved	RO	-

Table 3-279 por_cxla_por_cxla_aux_ctl (high) (continued)

Bits	Field name	Description	Type	Reset
38:36	misc_starv_th	Maximum number of consecutive instances a Misc message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
35	Reserved	Reserved	RO	-
34:32	snp_starv_th	Maximum number of consecutive instances a Snoop Request message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010

The following image shows the lower register bit assignments.

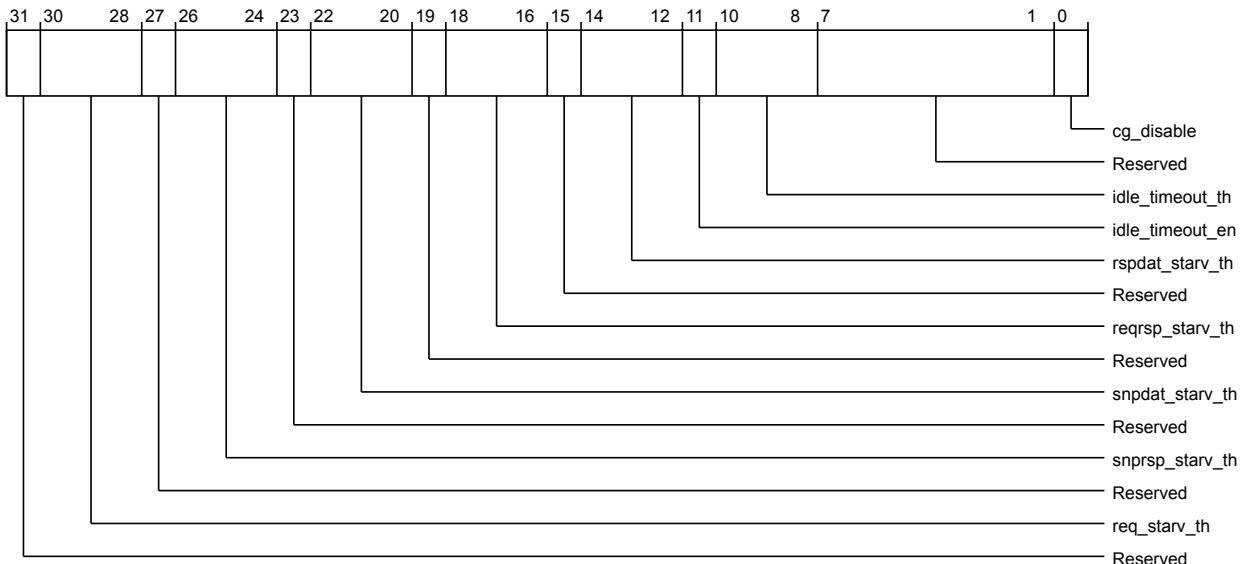


Figure 3-260 por_cxla_por_cxla_aux_ctl (low)

The following table shows the por_cxla_aux_ctl lower register bit assignments.

Table 3-280 por_cxla_por_cxla_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	req_starv_th	Maximum number of consecutive instances a Memory Request message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
27	Reserved	Reserved	RO	-
26:24	snprsp_starv_th	Maximum number of consecutive instances a Snoop Response without Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
23	Reserved	Reserved	RO	-
22:20	snpdat_starv_th	Maximum number of consecutive instances a Snoop Response with Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
19	Reserved	Reserved	RO	-
18:16	reqrsp_starv_th	Maximum number of consecutive instances a Memory Response without Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
15	Reserved	Reserved	RO	-

Table 3-280 por_cxla_por_cxla_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
14:12	rspdat_starv_th	Maximum number of consecutive instances a Memory Response with Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
11	idle_timeout_en	Enables idle timeout; applies for message packing When this bit is set, TLP packing continues until TLP length reaches the maximum configured or if the idle_timeout_th is reached	RW	1'b0
10:8	idle_timeout_th	Maximum number of idle cycles a TLP waits for a message to be packed before ending the TLP; applies for message packing 3'b000: 4 cycles 3'b001: 8 cycles 3'b010: 16 cycles 3'b011: 32 cycles	RW	3'b001
7:1	Reserved	Reserved	RO	-
0	cg_disable	Disables CXLA architectural clock gates	RW	1'b0

por_cxla_ccix_prop_capabilities

Contains CCIX-supported properties.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hC00

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

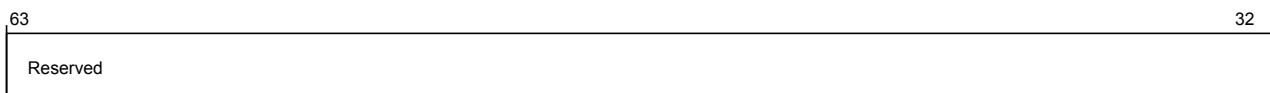


Figure 3-261 por_cxla_por_cxla_ccix_prop_capabilities (high)

The following table shows the por_cxla_ccix_prop_capabilities higher register bit assignments.

Table 3-281 por_cxla_por_cxla_ccix_prop_capabilities (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

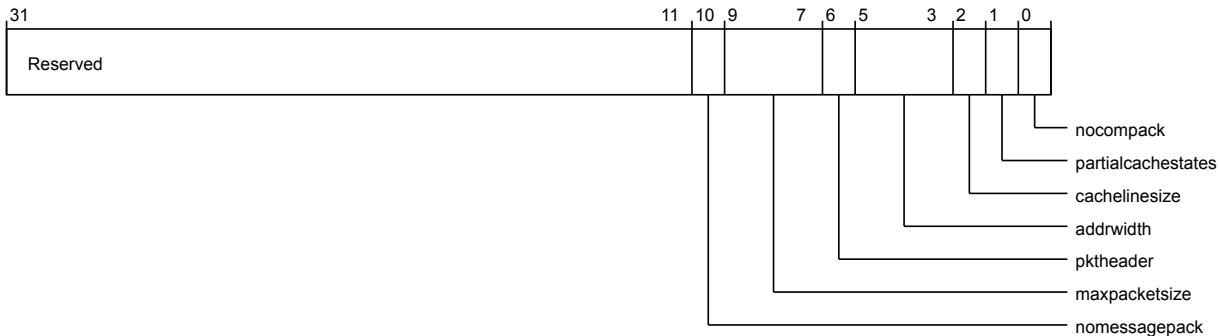


Figure 3-262 por_cxla_por_cxla_ccix_prop_capabilities (low)

The following table shows the por_exla_ccix_prop_capabilities lower register bit assignments.

Table 3-282 por_cxla_por_cxla_ccix_prop_capabilities (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10	nomessagepack	No message packing only supported 1'b0: False 1'b1: True	RO	Configuration dependent
9:7	maxpacketsize	Maximum packet size supported 3'b000: 128B 3'b001: 256B 3'b010: 512B	RO	Configuration dependent
6	pktheader	Packet header supported 1'b0: PCIe compatible header 1'b1: Optimized header	RO	Configuration dependent
5:3	addrwidth	Address width supported 3'b000: 48b 3'b001: 52b 3'b010: 56b 3'b011: 60b 3'b100: 64b	RO	Configuration dependent

Table 3-282 por_cxla_por_cxla_ccix_prop_capabilities (low) (continued)

Bits	Field name	Description	Type	Reset
2	cachelinesize	Cacheline size supported 1'b0: 64B 1'b1: 128B	RO	Configuration dependent
1	partialcachestates	Partial cache states supported 1'b0: False 1'b1: True	RO	Configuration dependent
0	nocompack	No CompAck supported 1'b0: False 1'b1: True	RO	Configuration dependent

por_cxla_ccix_prop_configured

Contains CCIX-configured properties.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC08

Register reset 64'b10000001000

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

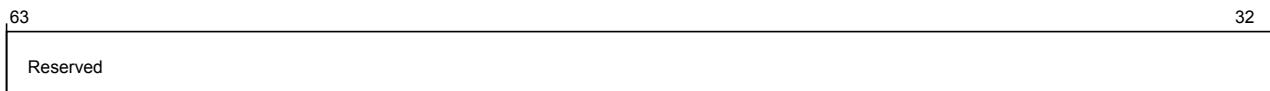


Figure 3-263 por_cxla_por_cxla_ccix_prop_configured (high)

The following table shows the por_cxla_ccix_prop_configured higher register bit assignments.

Table 3-283 por_cxla_por_cxla_ccix_prop_configured (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

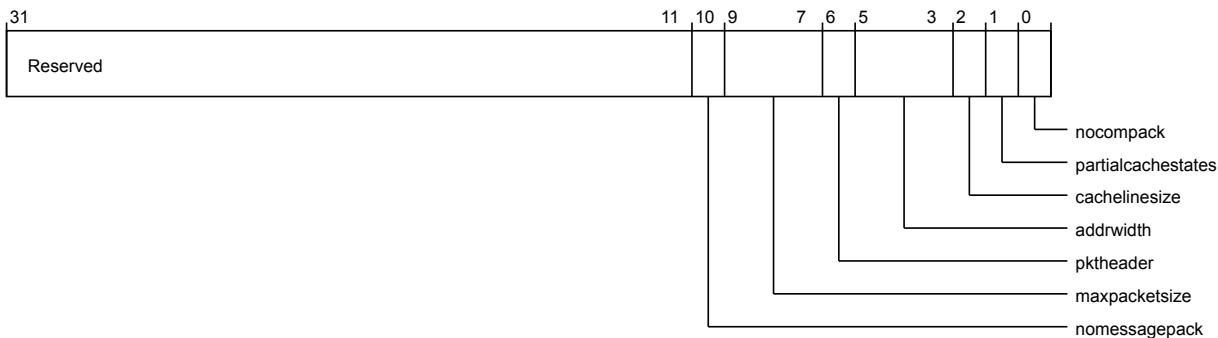


Figure 3-264 por_cxla_por_cxla_ccix_prop_configured (low)

The following table shows the por cxla ccix prop configured lower register bit assignments.

Table 3-284 por_cxla_por_cxla_ccix_prop_configured (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10	nomessagepack	No message packing configured 1'b0: False 1'b1: True	RW	1'b1
9:7	maxpacketsize	Maximum packet size configured 3'b000: 128B 3'b001: 256B 3'b010: 512B	RW	3'b000
6	pktheader	Packet header configured 1'b0: PCIe compatible header 1'b1: Optimized header	RW	1'b0
5:3	addrwidth	Address width configured 3'b000: 48b 3'b001: 52b 3'b010: 56b 3'b011: 60b 3'b100: 64b	RW	3'b001
2	cachelinesize	CacheLine size configured 1'b0: 64B 1'b1: 128B	RW	1'b0

Table 3-284 por_cxla_por_cxla_ccix_prop_configured (low) (continued)

Bits	Field name	Description	Type	Reset
1	partialcachestates	Partial cache states configured 1'b0: False 1'b1: True	RW	1'b0
0	nocompack	No CompAck configured 1'b0: False 1'b1: True	RW	1'b0

por_cxla_tx_cxs_attr_capabilities

Contains TX CXS supported attributes.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hC10

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

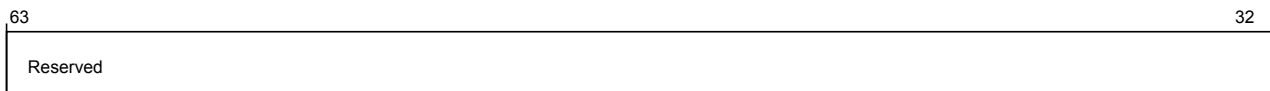


Figure 3-265 por_cxla_por_cxla_tx_cxs_attr_capabilities (high)

The following table shows the por_cxla_tx_cxs_attr_capabilities higher register bit assignments.

Table 3-285 por_cxla_por_cxla_tx_cxs_attr_capabilities (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

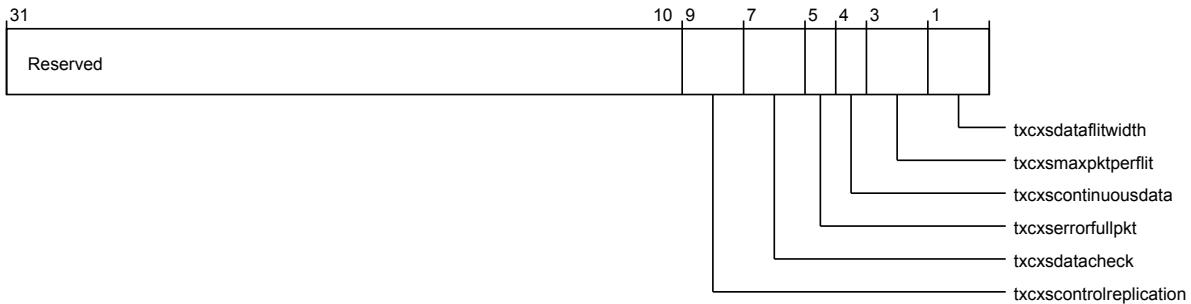


Figure 3-266 por_cxla_por_cxla_tx_cxs_attr_capabilities (low)

The following table shows the por_cxla_tx_cxs_attr_capabilities lower register bit assignments.

Table 3-286 por_cxla_por_cxla_tx_cxs_attr_capabilities (low)

Bits	Field name	Description	Type	Reset
31:10	Reserved	Reserved	RO	-
9:8	txcxscontrolreplication	TX CXS control replication supported 2'b00: None 2'b01: Duplicate 2'b10: Triuplicate	RO	Configuration dependent
7:6	txcxsdatabatch	TX CXS databatch supported 2'b00: None 2'b01: Parity 2'b10: SECDED	RO	Configuration dependent
5	txcxsserrorfullpkt	TX CXS error full packet supported 1'b0: False 1'b1: True	RO	Configuration dependent
4	txcxscardinuousdata	TX CXS continuous data supported 1'b0: False 1'b1: True	RO	Configuration dependent
3:2	txcxsmaxpktperflit	TX CXS maximum packets per flit supported 2'b00: 2 2'b01: 3 2'b10: 4	RO	Configuration dependent
1:0	txcxsdatalitwidth	TX CXS data flit width supported 2'b00: 256b 2'b01: 512b 2'b10: 1024b	RO	Configuration dependent

por_cxla_rx_cxs_attr_capabilities

Contains RX CXS supported attributes.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hC18

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-267 por_cxla_por_cxla_rx_cxs_attr_capabilities (high)

The following table shows the por_cxla_rx_cxs_attr_capabilities higher register bit assignments.

Table 3-287 por_cxla_por_cxla_rx_cxs_attr_capabilities (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

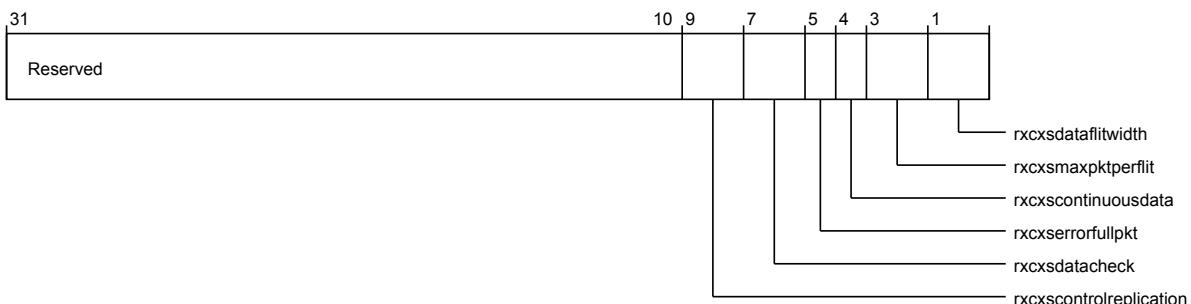


Figure 3-268 por_cxla_por_cxla_rx_cxs_attr_capabilities (low)

The following table shows the por_cxla_rx_cxs_attr_capabilities lower register bit assignments.

Table 3-288 por_cxla_por_cxla_rx_cxs_attr_capabilities (low)

Bits	Field name	Description	Type	Reset
31:10	Reserved	Reserved	RO	-
9:8	rxcxscontrolreplication	RX CXS control replication supported 2'b00: None 2'b01: Duplicate 2'b10: Triuplicate	RO	Configuration dependent
7:6	rxcxsdatacheck	RX CXS datacheck supported 2'b00: None 2'b01: Parity 2'b10: SECDED	RO	Configuration dependent
5	rxcxSErrorfullpkt	RX CXS error full packet supported 1'b0: False 1'b1: True	RO	Configuration dependent
4	rxcxscontinuousdata	RX CXS continuous data supported 1'b0: False 1'b1: True	RO	Configuration dependent
3:2	rxcxsmaxpktperflit	RX CXS maximum packets per flit supported 2'b00: 2 2'b01: 3 2'b10: 4	RO	Configuration dependent
1:0	rxcxldataflitwidth	RX CXS data flit width supported 2'b00: 256b 2'b01: 512b 2'b10: 1024b	RO	Configuration dependent

por_cxla_agentid_to_linkid_reg0

Specifies the mapping of Agent ID to Link ID for Agent IDs 0 to 7.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC30

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxla_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

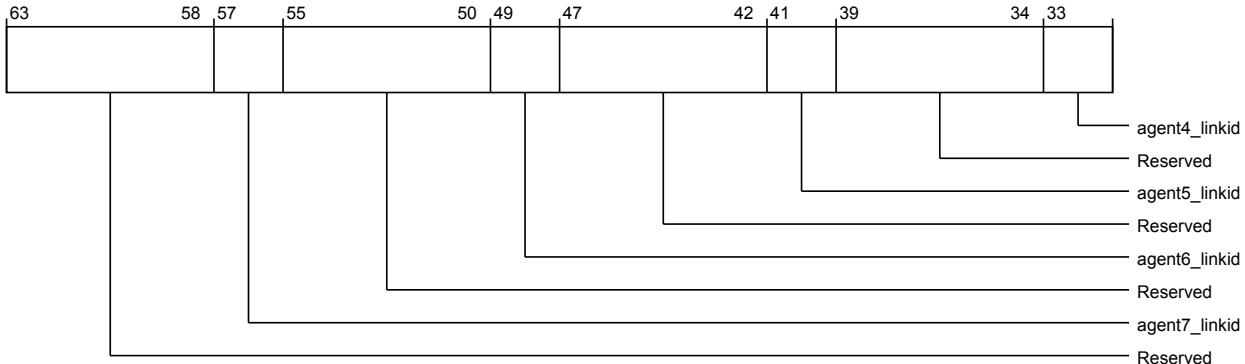


Figure 3-269 por_cxla_por_cxla_agentid_to_linkid_reg0 (high)

The following table shows the port cxla agentid to linkid reg0 higher register bit assignments.

Table 3-289 por_cxla_por_cxla_agentid_to_linkid_reg0 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent7_linkid	Specifies the Link ID for Agent ID 7	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent6_linkid	Specifies the Link ID for Agent ID 6	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent5_linkid	Specifies the Link ID for Agent ID 5	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent4_linkid	Specifies the Link ID for Agent ID 4	RW	2'h0

The following image shows the lower register bit assignments.

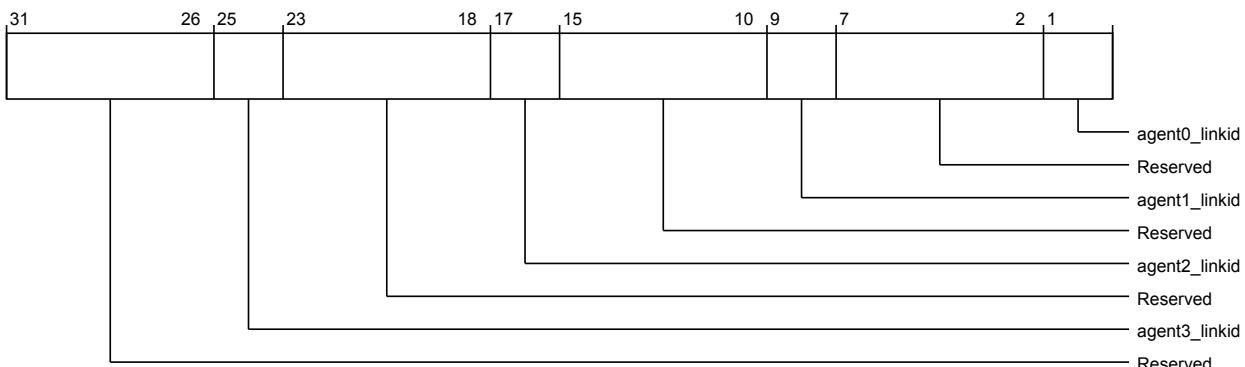


Figure 3-270 por_cxla_por_cxla_agentid_to_linkid_reg0 (low)

The following table shows the por cxla agentid to linkid reg0 lower register bit assignments.

Table 3-290 por_cxla_por_cxla_agentid_to_linkid_reg0 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent3_linkid	Specifies the Link ID for Agent ID 3	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent2_linkid	Specifies the Link ID for Agent ID 2	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent1_linkid	Specifies the Link ID for Agent ID 1	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent0_linkid	Specifies the Link ID for Agent ID 0	RW	2'h0

por_cxla_agentid_to_linkid_reg1

Specifies the mapping of Agent ID to Link ID for Agent IDs 8 to 15.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC38

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxla_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

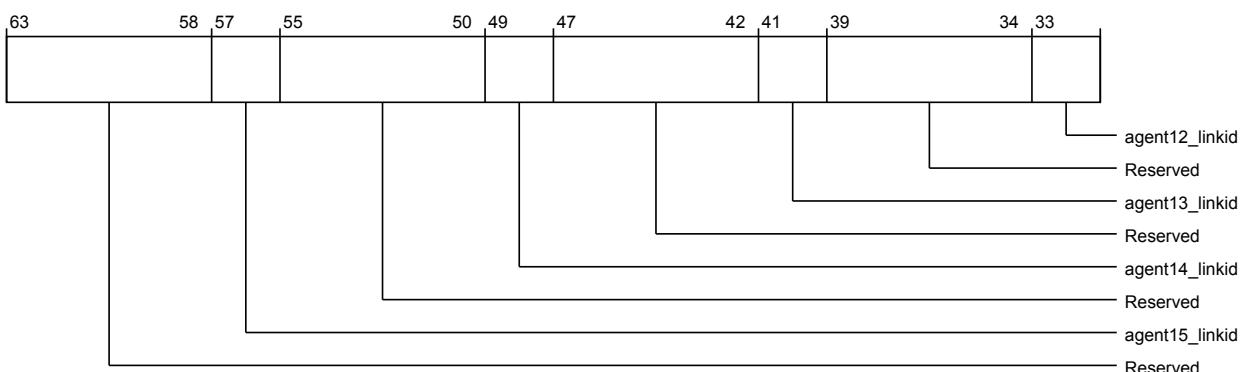


Figure 3-271 por_cxla_por_cxla_agentid_to_linkid_reg1 (high)

The following table shows the por_cxla_agentid_to_linkid_reg1 higher register bit assignments.

Table 3-291 por_cxla_por_cxla_agentid_to_linkid_reg1 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent15_linkid	Specifies the Link ID for Agent ID 15	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent14_linkid	Specifies the Link ID for Agent ID 14	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent13_linkid	Specifies the Link ID for Agent ID 13	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent12_linkid	Specifies the Link ID for Agent ID 12	RW	2'h0

The following image shows the lower register bit assignments.

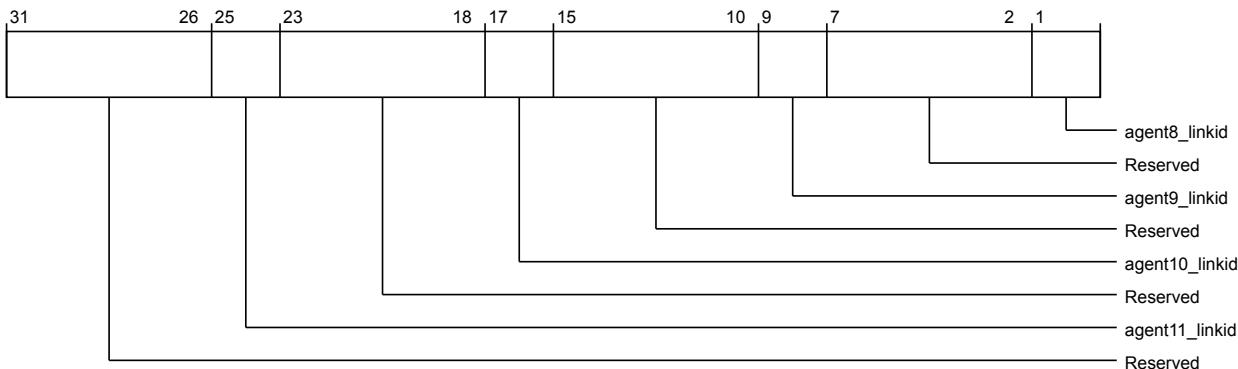


Figure 3-272 por_cxla_por_cxla_agentid_to_linkid_reg1 (low)

The following table shows the por_cxla_agentid_to_linkid_reg1 lower register bit assignments.

Table 3-292 por_cxla_por_cxla_agentid_to_linkid_reg1 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent11_linkid	Specifies the Link ID for Agent ID 11	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent10_linkid	Specifies the Link ID for Agent ID 10	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent9_linkid	Specifies the Link ID for Agent ID 9	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent8_linkid	Specifies the Link ID for Agent ID 8	RW	2'h0

por_cxla_agentid_to_linkid_reg2

Specifies the mapping of Agent ID to Link ID for Agent IDs 16 to 23.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxla_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

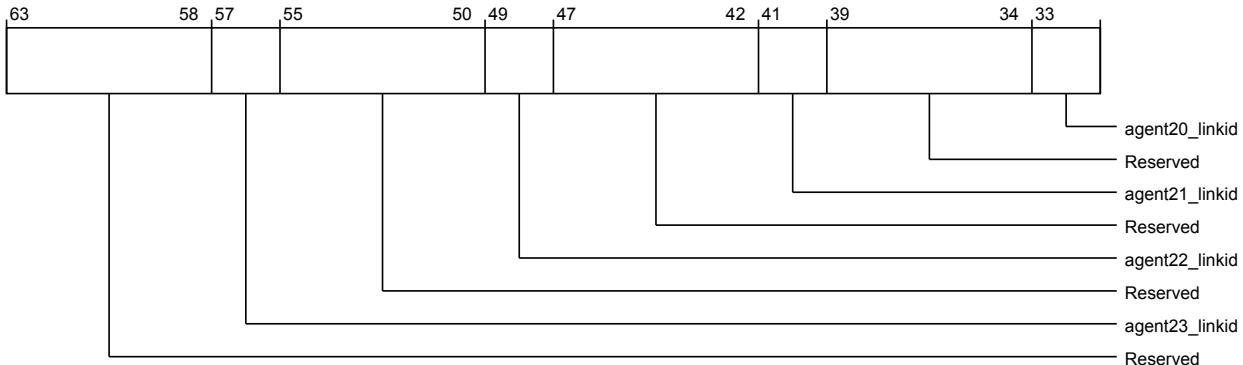


Figure 3-273 por_cxla_por_cxla_agentid_to_linkid_reg2 (high)

The following table shows the port cxla agentid to linkid reg2 higher register bit assignments.

Table 3-293 por_cxla_por_cxla_agentid_to_linkid_reg2 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent23_linkid	Specifies the Link ID for Agent ID 23	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent22_linkid	Specifies the Link ID for Agent ID 22	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent21_linkid	Specifies the Link ID for Agent ID 21	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent20_linkid	Specifies the Link ID for Agent ID 20	RW	2'h0

The following image shows the lower register bit assignments.

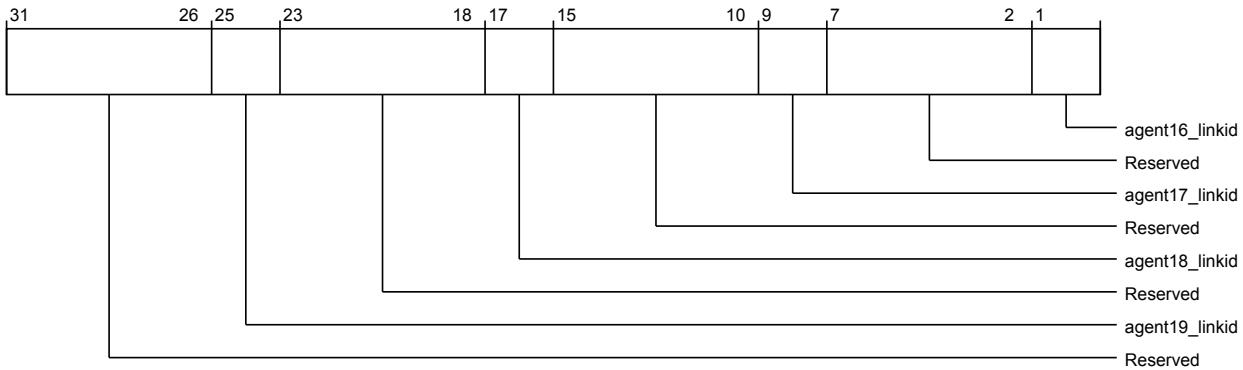


Figure 3-274 por_cxla_por_cxla_agentid_to_linkid_reg2 (low)

The following table shows the por_cxla_agentid to linkid_reg2 lower register bit assignments.

Table 3-294 por_cxla_por_cxla_agentid_to_linkid_reg2 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent19_linkid	Specifies the Link ID for Agent ID 19	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent18_linkid	Specifies the Link ID for Agent ID 18	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent17_linkid	Specifies the Link ID for Agent ID 17	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent16_linkid	Specifies the Link ID for Agent ID 16	RW	2'h0

por_cxla_agentid_to_linkid_reg3

Specifies the mapping of Agent ID to Link ID for Agent IDs 24 to 31.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC3

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_cxla_secure_register_groups_override.linkid_ctl

The following sections describe the high-level architecture of the system.

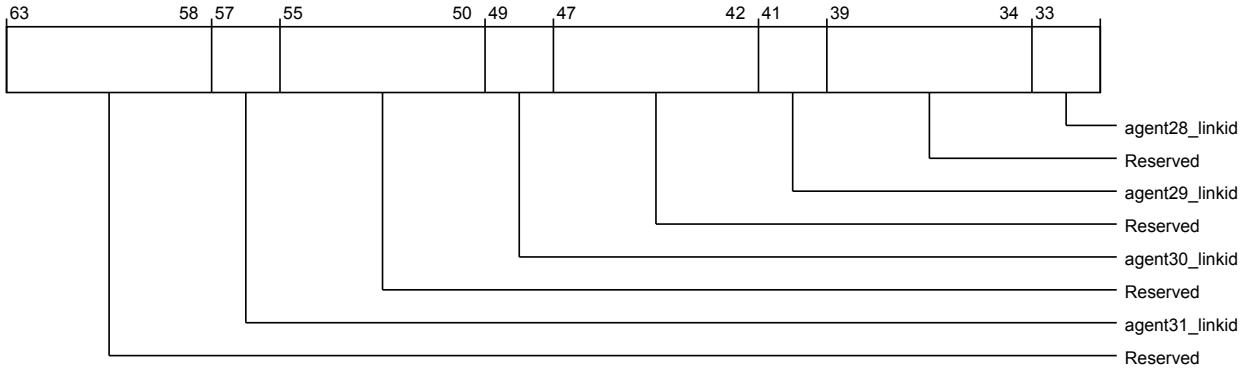


Figure 3-275 por_cxla_por_cxla_agentid_to_linkid_reg3 (high)

The following table shows the por_cxla_agentid to linkid_reg3 higher register bit assignments.

Table 3-295 por_cxla_por_cxla_agentid_to_linkid_reg3 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent31_linkid	Specifies the Link ID for Agent ID 31	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent30_linkid	Specifies the Link ID for Agent ID 30	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent29_linkid	Specifies the Link ID for Agent ID 29	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent28_linkid	Specifies the Link ID for Agent ID 28	RW	2'h0

The following image shows the lower register bit assignments.

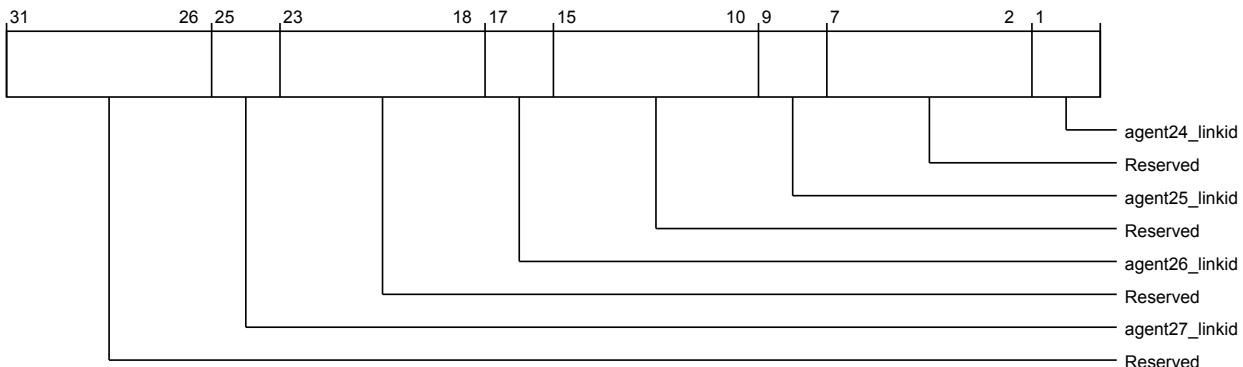


Figure 3-276 por_cxla_por_cxla_agentid_to_linkid_reg3 (low)

The following table shows the port cxla agentid to linkid reg3 lower register bit assignments.

Table 3-296 por_cxla_por_cxla_agentid_to_linkid_reg3 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent27_linkid	Specifies the Link ID for Agent ID 27	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent26_linkid	Specifies the Link ID for Agent ID 26	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent25_linkid	Specifies the Link ID for Agent ID 25	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent24_linkid	Specifies the Link ID for Agent ID 24	RW	2'h0

por_cxla_agentid_to_linkid_reg4

Specifies the mapping of Agent ID to Link ID for Agent IDs 32 to 39.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC50

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxla_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

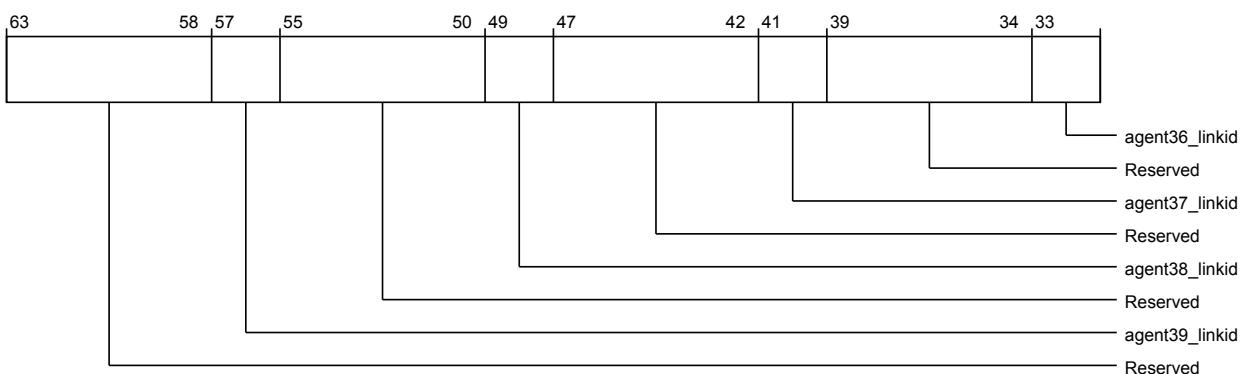


Figure 3-277 por_cxla_por_cxla_agentid_to_linkid_reg4 (high)

The following table shows the por_cxla_agentid_to_linkid_reg4 higher register bit assignments.

Table 3-297 por_cxla_por_cxla_agentid_to_linkid_reg4 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent39_linkid	Specifies the Link ID for Agent ID 39	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent38_linkid	Specifies the Link ID for Agent ID 38	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent37_linkid	Specifies the Link ID for Agent ID 37	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent36_linkid	Specifies the Link ID for Agent ID 36	RW	2'h0

The following image shows the lower register bit assignments.

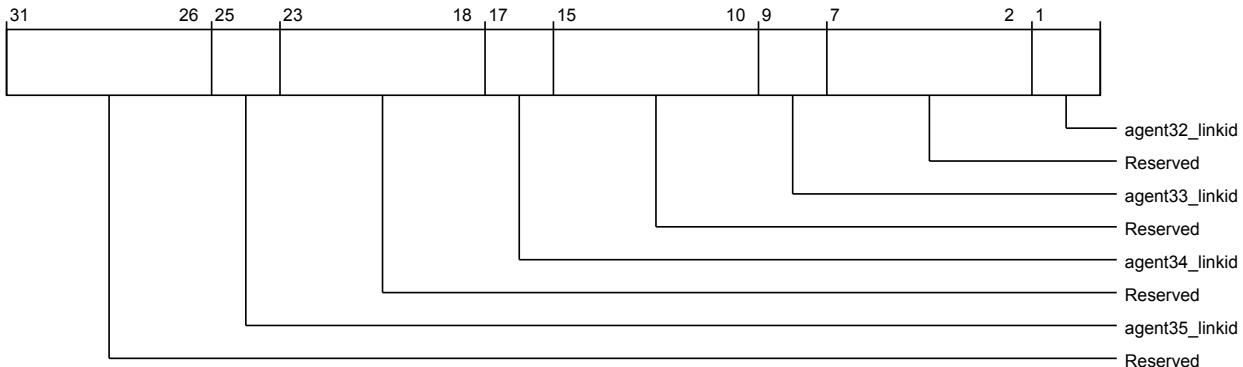


Figure 3-278 por_cxla_por_cxla_agentid_to_linkid_reg4 (low)

The following table shows the por_cxla_agentid_to_linkid_reg4 lower register bit assignments.

Table 3-298 por_cxla_por_cxla_agentid_to_linkid_reg4 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent35_linkid	Specifies the Link ID for Agent ID 35	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent34_linkid	Specifies the Link ID for Agent ID 34	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent33_linkid	Specifies the Link ID for Agent ID 33	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent32_linkid	Specifies the Link ID for Agent ID 32	RW	2'h0

por_cxla_agentid_to_linkid_reg5

Specifies the mapping of Agent ID to Link ID for Agent IDs 40 to 47.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC58
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxla_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

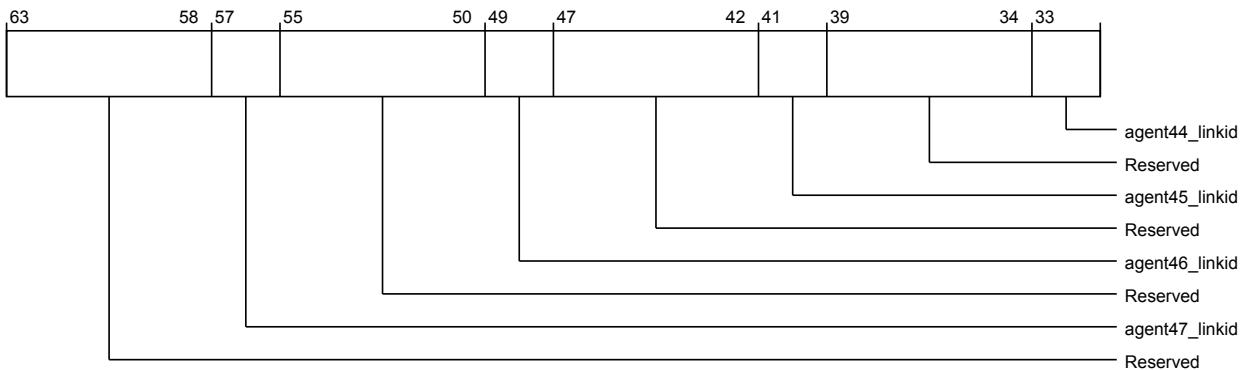


Figure 3-279 por_cxla_por_cxla_agentid_to_linkid_reg5 (high)

The following table shows the por_cxla_agentid_to_linkid_reg5 higher register bit assignments.

Table 3-299 por_cxla_por_cxla_agentid_to_linkid_reg5 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent47_linkid	Specifies the Link ID for Agent ID 47	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent46_linkid	Specifies the Link ID for Agent ID 46	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent45_linkid	Specifies the Link ID for Agent ID 45	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent44_linkid	Specifies the Link ID for Agent ID 44	RW	2'h0

The following image shows the lower register bit assignments.

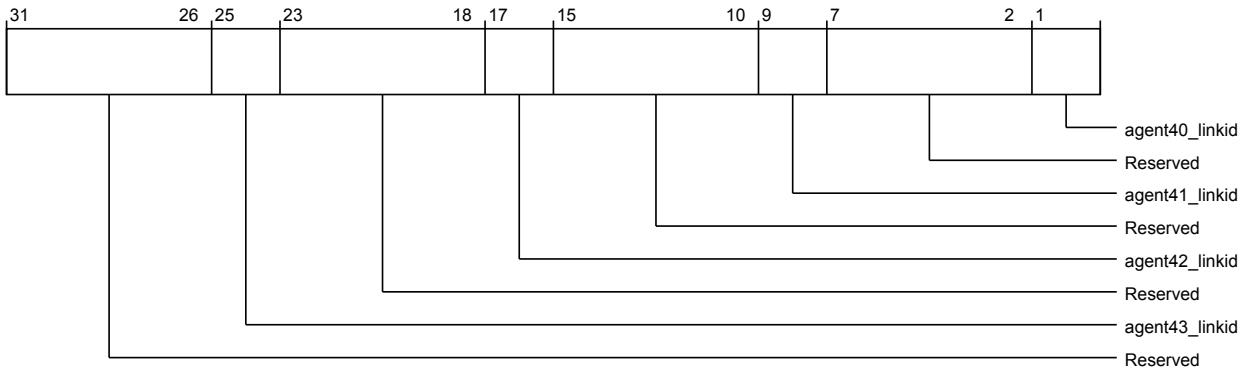


Figure 3-280 por_cxla_por_cxla_agentid_to_linkid_reg5 (low)

The following table shows the por_cxla_agentid to linkid_reg5 lower register bit assignments.

Table 3-300 por_cxla_por_cxla_agentid_to_linkid_reg5 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent43_linkid	Specifies the Link ID for Agent ID 43	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent42_linkid	Specifies the Link ID for Agent ID 42	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent41_linkid	Specifies the Link ID for Agent ID 41	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent40_linkid	Specifies the Link ID for Agent ID 40	RW	2'h0

por_cxla_agentid_to_linkid_reg6

Specifies the mapping of Agent ID to Link ID for Agent IDs 48 to 55.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCe

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_cxla_secure_register_groups_override.linkid_ctl

The following sections describe the high-level architecture of the system.

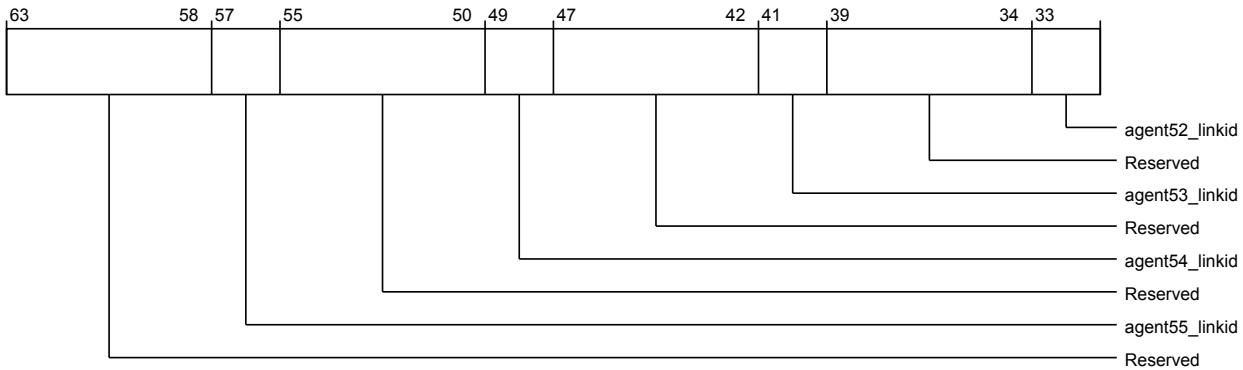


Figure 3-281 por_cxla_por_cxla_agentid_to_linkid_reg6 (high)

The following table shows the por_cxla_agentid_to_linkid_reg6 higher register bit assignments.

Table 3-301 por_cxla_por_cxla_agentid_to_linkid_reg6 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent55_linkid	Specifies the Link ID for Agent ID 55	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent54_linkid	Specifies the Link ID for Agent ID 54	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent53_linkid	Specifies the Link ID for Agent ID 53	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent52_linkid	Specifies the Link ID for Agent ID 52	RW	2'h0

The following image shows the lower register bit assignments.

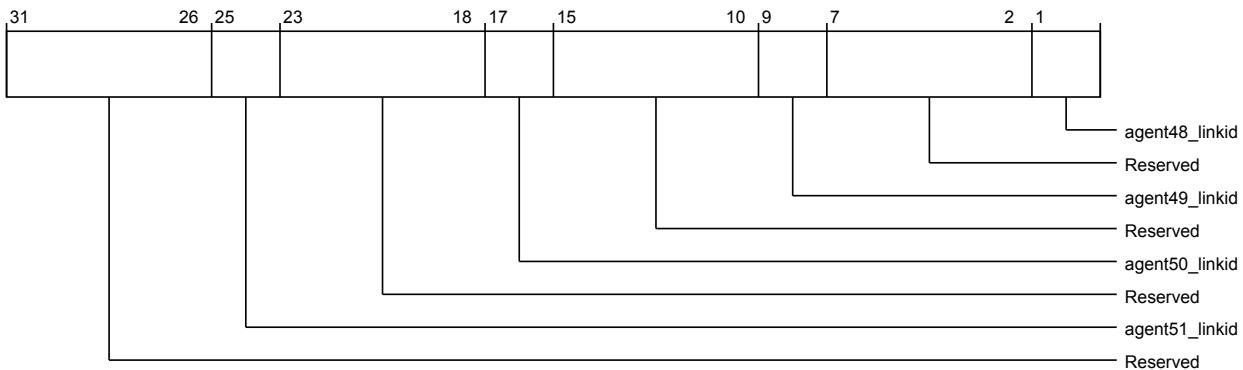


Figure 3-282 por_cxla_por_cxla_agentid_to_linkid_reg6 (low)

The following table shows the por_cxla_agentid_to_linkid_reg6 lower register bit assignments.

Table 3-302 por_cxla_por_cxla_agentid_to_linkid_reg6 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent51_linkid	Specifies the Link ID for Agent ID 51	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent50_linkid	Specifies the Link ID for Agent ID 50	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent49_linkid	Specifies the Link ID for Agent ID 49	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent48_linkid	Specifies the Link ID for Agent ID 48	RW	2'h0

por_cxla_agentid_to_linkid_reg7

Specifies the mapping of Agent ID to Link ID for Agent IDs 56 to 63.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC68

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxla_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

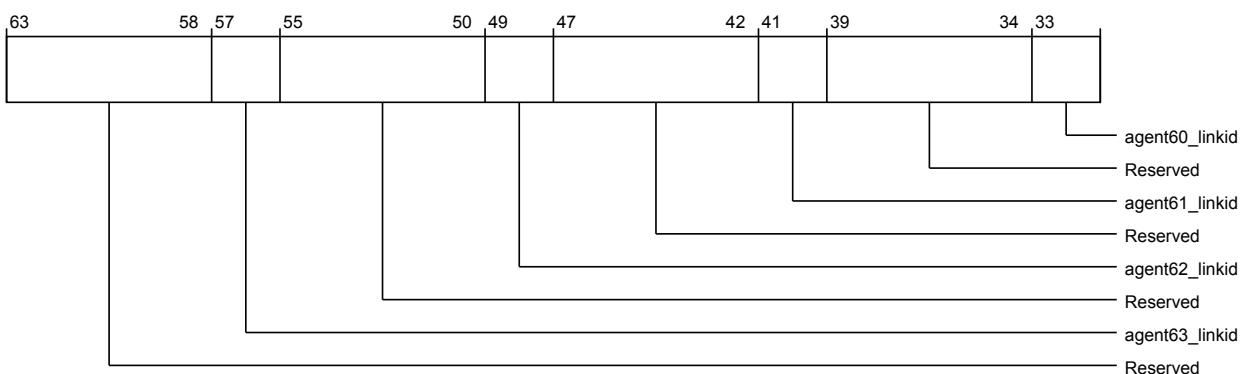


Figure 3-283 por_cxla_por_cxla_agentid_to_linkid_reg7 (high)

The following table shows the por_cxla_agentid_to_linkid_reg7 higher register bit assignments.

Table 3-303 por_cxla_por_cxla_agentid_to_linkid_reg7 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent63_linkid	Specifies the Link ID for Agent ID 63	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent62_linkid	Specifies the Link ID for Agent ID 62	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent61_linkid	Specifies the Link ID for Agent ID 61	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent60_linkid	Specifies the Link ID for Agent ID 60	RW	2'h0

The following image shows the lower register bit assignments.

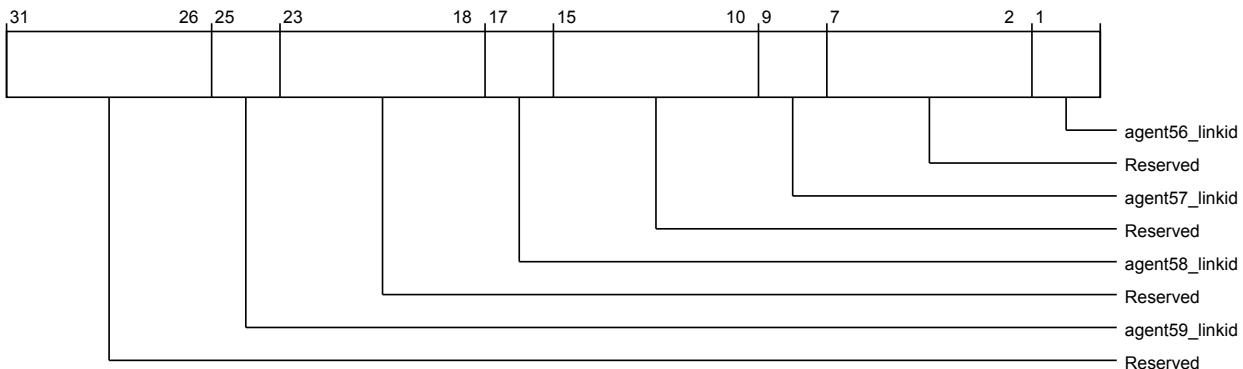


Figure 3-284 por_cxla_por_cxla_agentid_to_linkid_reg7 (low)

The following table shows the por_cxla_agentid_to_linkid_reg7 lower register bit assignments.

Table 3-304 por_cxla_por_cxla_agentid_to_linkid_reg7 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent59_linkid	Specifies the Link ID for Agent ID 59	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent58_linkid	Specifies the Link ID for Agent ID 58	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent57_linkid	Specifies the Link ID for Agent ID 57	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent56_linkid	Specifies the Link ID for Agent ID 56	RW	2'h0

por_cxla_agentid_to_linkid_val

Specifies which Agent ID to Link ID mappings are valid.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC70
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxla_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.



Figure 3-285 por_cxla_por_cxla_agentid_to_linkid_val (high)

The following table shows the por_cxla_agentid_to_linkid_val higher register bit assignments.

Table 3-305 por_cxla_por_cxla_agentid_to_linkid_val (high)

Bits	Field name	Description	Type	Reset
63:32	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

The following image shows the lower register bit assignments.

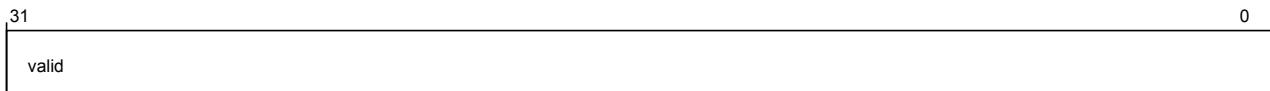


Figure 3-286 por_cxla_por_cxla_agentid_to_linkid_val (low)

The following table shows the por_cxla_agentid_to_linkid_val lower register bit assignments.

Table 3-306 por_cxla_por_cxla_agentid_to_linkid_val (low)

Bits	Field name	Description	Type	Reset
31:0	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

por_cxla_linkid_to_pcie_bus_num

Specifies the mapping of CCIX Link ID to PCIe bus number.

Its characteristics are:

Type	RW
-------------	----

Register width (Bits) 64

Address offset 16'hC78

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

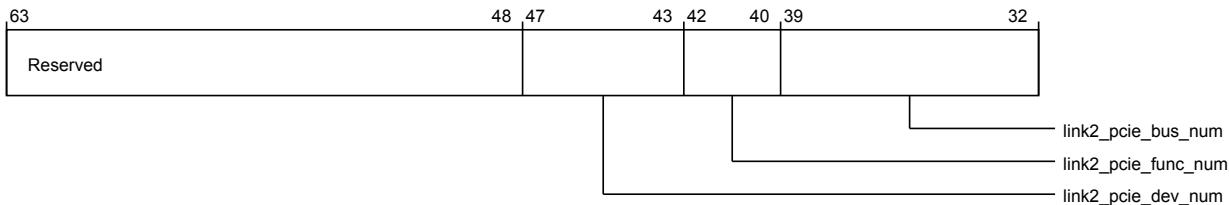


Figure 3-287 por_cxla_por_cxla_linkid_to_pcie_bus_num (high)

The following table shows the por_cxla_linkid_to_pcie_bus_num higher register bit assignments.

Table 3-307 por_cxla_por_cxla_linkid_to_pcie_bus_num (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:43	link2_pcie_dev_num	PCIe Device number for Link ID 2	RW	5'h00
42:40	link2_pcie_func_num	PCIe Function number for Link ID 2	RW	3'h0
39:32	link2_pcie_bus_num	PCIe bus number for Link ID 2	RW	8'h00

The following image shows the lower register bit assignments.

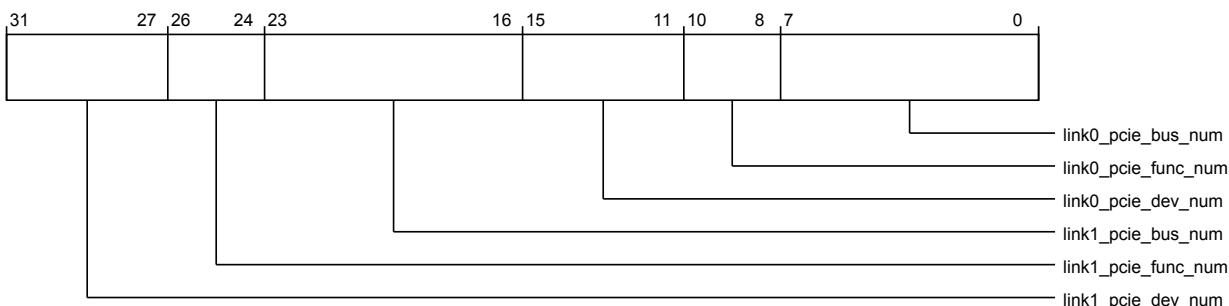


Figure 3-288 por_cxla_linkid_to_pcie_bus_num (low)

The following table shows the por_cxla_linkid_to_pcie_bus_num lower register bit assignments.

Table 3-308 por_cxla_linkid_to_pcie_bus_num (low)

Bits	Field name	Description	Type	Reset
31:27	link1_pcie_dev_num	PCIe Device number for Link ID 1	RW	5'h00
26:24	link1_pcie_func_num	PCIe Function number for Link ID 1	RW	3'h0

Table 3-308 por_cxla_por_cxla_linkid_to_pcie_bus_num (low) (continued)

Bits	Field name	Description	Type	Reset
23:16	link1_pcie_bus_num	PCIe bus number for Link ID 1	RW	8'h00
15:11	link0_pcie_dev_num	PCIe Device number for Link ID 0	RW	5'h00
10:8	link0_pcie_func_num	PCIe Function number for Link ID 0	RW	3'h0
7:0	link0_pcie_bus_num	PCIe bus number for Link ID 0	RW	8'h00

por_cxla_tlp_hdr_fields

Configures TLP header field values.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC80
Register reset	64'b001111110000000000001110010
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

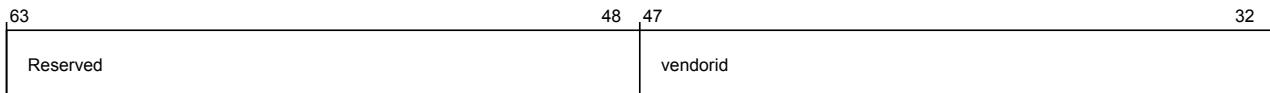


Figure 3-289 por_cxla_por_cxla_tlp_hdr_fields (high)

The following table shows the por_cxla_tlp_hdr_fields higher register bit assignments.

Table 3-309 por_cxla_por_cxla_tlp_hdr_fields (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	vendorid	Vendor ID	RW	16'b0

The following image shows the lower register bit assignments.

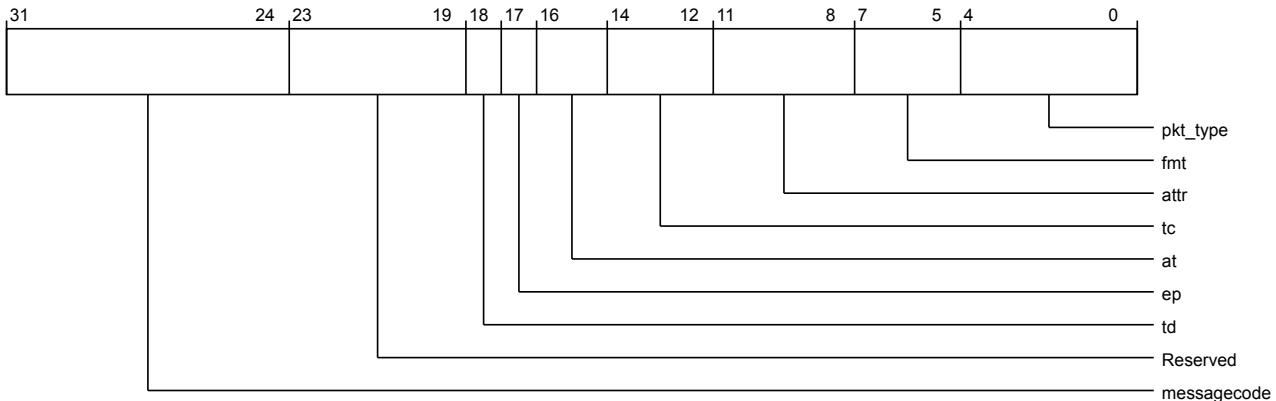


Figure 3-290 por_cxla_por_cxla_tlp_hdr_fields (low)

The following table shows the por_cxla_tlp_hdr_fields lower register bit assignments.

Table 3-310 por_cxla_por_cxla_tlp_hdr_fields (low)

Bits	Field name	Description	Type	Reset
31:24	messagecode	Message code	RW	8'b01111111
23:19	Reserved	Reserved	RO	-
18	td	TLP digest	RW	1'b0
17	ep	Error forwarding	RW	1'b0
16:15	at	Address type	RW	2'b00
14:12	tc	Traffic class	RW	3'b000
11:8	attr	Attributes	RW	4'b0000
7:5	fmt	Format	RW	3'b011
4:0	pkt_type	Type	RW	5'b10010

por_cxla_permsg_pyld_0_63

Contains bits[63:0] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD00

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

per_msg_pyld_0_63

Figure 3-291 por_cxla_por_cxla_permsg_pyld_0_63 (high)

The following table shows the por_cxla_permsg_pyld_0_63 higher register bit assignments.

Table 3-311 por_cxla_por_cxla_permsg_pyld_0_63 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_0_63	Protocol Error Msg Payload[63:0]	RW	64'b0

The following image shows the lower register bit assignments.

31

0

per_msg_pyld_0_63

Figure 3-292 por_cxla_por_cxla_permsg_pyld_0_63 (low)

The following table shows the por_cxla_permsg_pyld_0_63 lower register bit assignments.

Table 3-312 por_cxla_por_cxla_permsg_pyld_0_63 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_0_63	Protocol Error Msg Payload[63:0]	RW	64'b0

por_cxla_permsg_pyld_64_127

Contains bits[127:64] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD08

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

per_msg_pyld_64_127

Figure 3-293 por_cxla_por_cxla_permsg_pyld_64_127 (high)

The following table shows the por_cxla_permsg_pyld_64_127 higher register bit assignments.

Table 3-313 por_cxla_por_cxla_permmsg_pyld_64_127 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_64_127	Protocol Error Msg Payload[127:64]	RW	64'b0

The following image shows the lower register bit assignments.

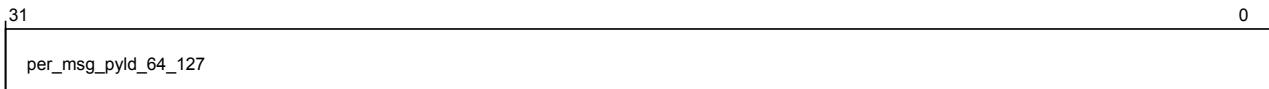


Figure 3-294 por_cxla_por_cxla_permmsg_pyld_64_127 (low)

The following table shows the por_cxla_permmsg_pyld_64_127 lower register bit assignments.

Table 3-314 por_cxla_por_cxla_permmsg_pyld_64_127 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_64_127	Protocol Error Msg Payload[127:64]	RW	64'b0

por_cxla_permmsg_pyld_128_191

Contains bits[192:128] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD10

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

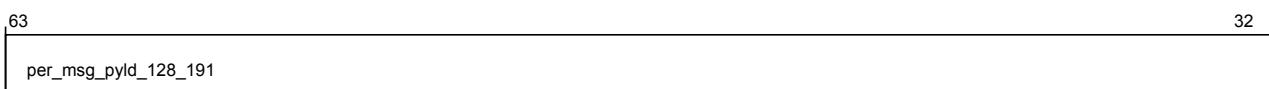


Figure 3-295 por_cxla_por_cxla_permmsg_pyld_128_191 (high)

The following table shows the por_cxla_permmsg_pyld_128_191 higher register bit assignments.

Table 3-315 por_cxla_por_cxla_permmsg_pyld_128_191 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_128_191	Protocol Error Msg Payload[191:128]	RW	64'b0

The following image shows the lower register bit assignments.

31	per_msg_pyld_128_191	0
----	----------------------	---

Figure 3-296 por_cxla_por_cxla_permsg_pyld_128_191 (low)

The following table shows the por_cxla_permsg_pyld_128_191 lower register bit assignments.

Table 3-316 por_cxla_por_cxla_permsg_pyld_128_191 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_128_191	Protocol Error Msg Payload[191:128]	RW	64'b0

por_cxla_permsg_pyld_192_255

Contains bits[255:192] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD18

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	per_msg_pyld_192_255	32
----	----------------------	----

Figure 3-297 por_cxla_por_cxla_permsg_pyld_192_255 (high)

The following table shows the por_cxla_permsg_pyld_192_255 higher register bit assignments.

Table 3-317 por_cxla_por_cxla_permsg_pyld_192_255 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_192_255	Protocol Error Msg Payload[255:192]	RW	64'b0

The following image shows the lower register bit assignments.

31	per_msg_pyld_192_255	0
----	----------------------	---

Figure 3-298 por_cxla_por_cxla_permsg_pyld_192_255 (low)

The following table shows the por_cxla_permsg_pyld_192_255 lower register bit assignments.

Table 3-318 por_cxla_por_cxla_permsg_pyld_192_255 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_192_255	Protocol Error Msg Payload[255:192]	RW	64'b0

por_cxla_permsg_ctl

Contains Control bits to trigger CCIX Protocol Error (PER) Message.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD20

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

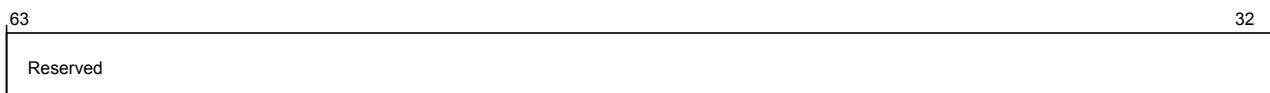


Figure 3-299 por_cxla_por_cxla_permsg_ctl (high)

The following table shows the por_cxla_permsg_ctl higher register bit assignments.

Table 3-319 por_cxla_por_cxla_permsg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

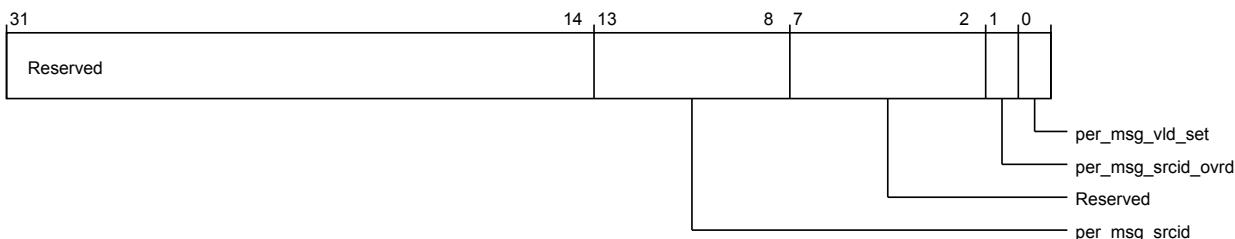


Figure 3-300 por_cxla_por_cxla_permsg_ctl (low)

The following table shows the por_cxla_permsg_ctl lower register bit assignments.

Table 3-320 por_cxla_por_cxla_permmsg_ctl (low)

Bits	Field name	Description	Type	Reset
31:14	Reserved	Reserved	RO	-
13:8	per_msg_srcid	Contains Source ID used on CCIX Protocol Error Msg. Used when per_msg_srcid_ovrd is set.	RW	6'b0
7:2	Reserved	Reserved	RO	-
1	per_msg_srcid_ovrd	When set, overrides the Source ID on Protocol Error Msg by value specified in this register. Or else the source ID from payload[55:48] is used.	RW	1'b0
0	per_msg_vld_set	When set, sends CCIX Protocol Error Msg. Must be cleared after the current error is processed and before a new error message is triggered	RW	1'b0

por_cxla_err_agent_id

Contains Error Agent ID. Must be programmed by CCIX discovery s/w. Used as TargetID on CCIX Protocol Error (PER) Message.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD28

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-301 por_cxla_por_cxla_err_agent_id (high)

The following table shows the por_cxla_err_agent_id higher register bit assignments.

Table 3-321 por_cxla_por_cxla_err_agent_id (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

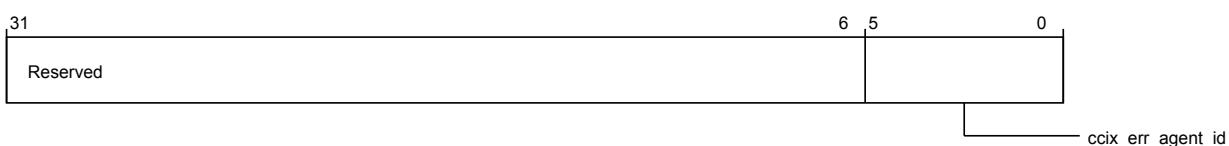


Figure 3-302 por_cxla_por_cxla_err_agent_id (low)

The following table shows the por_cxla_err_agent_id lower register bit assignments.

Table 3-322 por_cxla_por_cxla_err_agent_id (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5:0	ccix_err_agent_id	CCIX Error AgentID	RW	6'b0

por_cxla_agentid_to_portid_reg0

Specifies the mapping of Agent ID to Port ID for Agent IDs 0 to 7.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_cxla_secure_register_groups_override.portid_ctl

Table 3.11 A comparison of the 1990-1991 and 1991-1992 seasons.

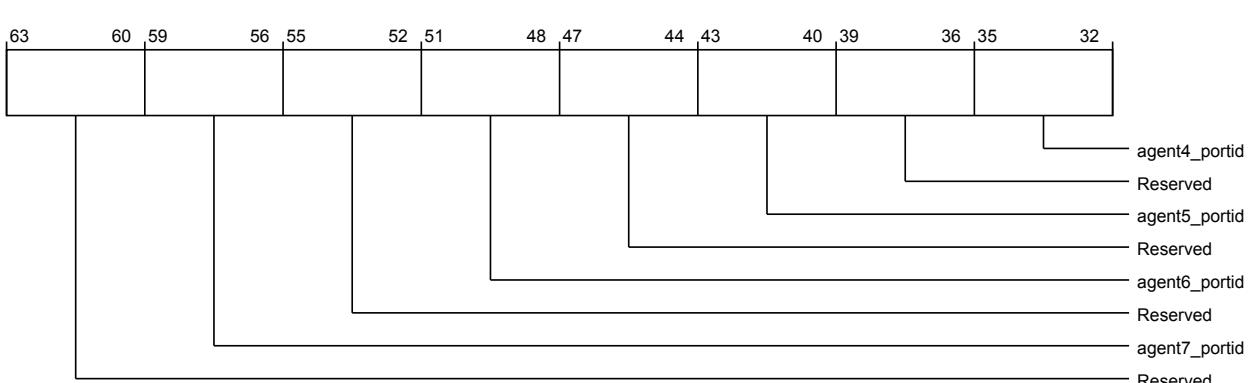


Figure 3-303 por cxla por cxla agentid to portid req0 (high)

The following table shows the portid to reg0 higher register bit assignments.

Table 3-323 por cxla por cxla agentid to portid req0 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent7_portid	Specifies the Port ID for Agent ID 7	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent6_portid	Specifies the Port ID for Agent ID 6	RW	4'h0
47:44	Reserved	Reserved	RO	-

Table 3-323 por_cxla_por_cxla_agentid_to_portid_reg0 (high) (continued)

Bits	Field name	Description	Type	Reset
43:40	agent5_portid	Specifies the Port ID for Agent ID 5	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent4_portid	Specifies the Port ID for Agent ID 4	RW	4'h0

The following image shows the lower register bit assignments.

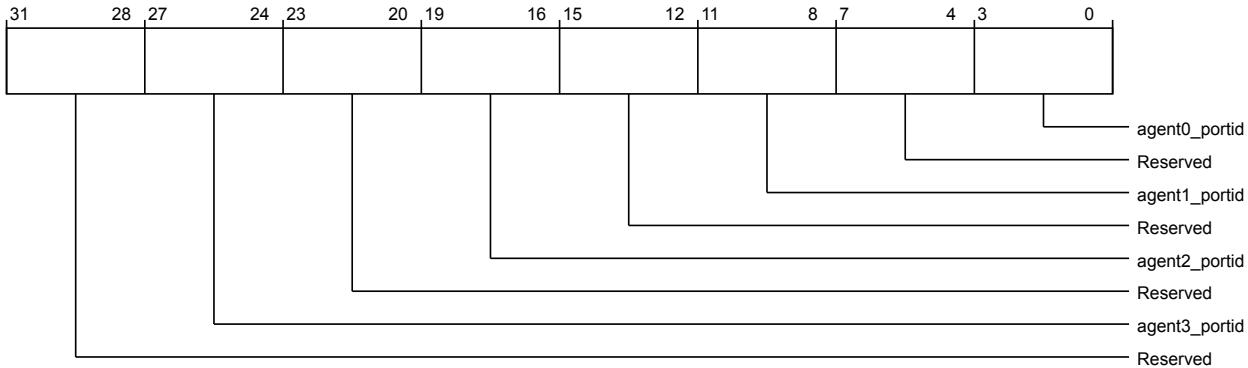


Figure 3-304 por_cxla_por_cxla_agentid_to_portid_reg0 (low)

The following table shows the por_cxla_agentid_to_portid_reg0 lower register bit assignments.

Table 3-324 por_cxla_por_cxla_agentid_to_portid_reg0 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent3_portid	Specifies the Port ID for Agent ID 3	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent2_portid	Specifies the Port ID for Agent ID 2	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent1_portid	Specifies the Port ID for Agent ID 1	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent0_portid	Specifies the Port ID for Agent ID 0	RW	4'h0

por_cxla_agentid_to_portid_reg1

Specifies the mapping of Agent ID to Port ID for Agent IDs 8 to 15.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD38

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

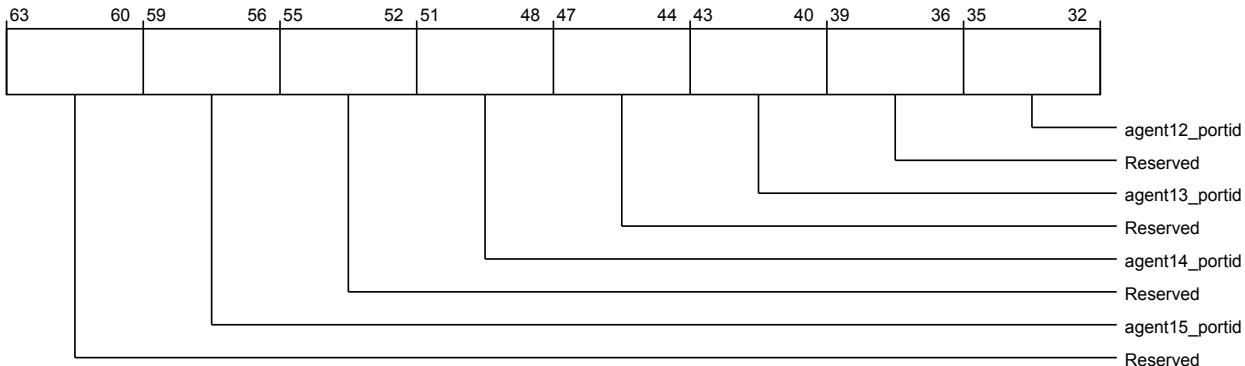


Figure 3-305 por_cxla_por_cxla_agentid_to_portid_reg1 (high)

The following table shows the por_cxla_agentid_to_portid_reg1 higher register bit assignments.

Table 3-325 por_cxla_por_cxla_agentid_to_portid_reg1 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent15_portid	Specifies the Port ID for Agent ID 15	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent14_portid	Specifies the Port ID for Agent ID 14	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent13_portid	Specifies the Port ID for Agent ID 13	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent12_portid	Specifies the Port ID for Agent ID 12	RW	4'h0

The following image shows the lower register bit assignments.

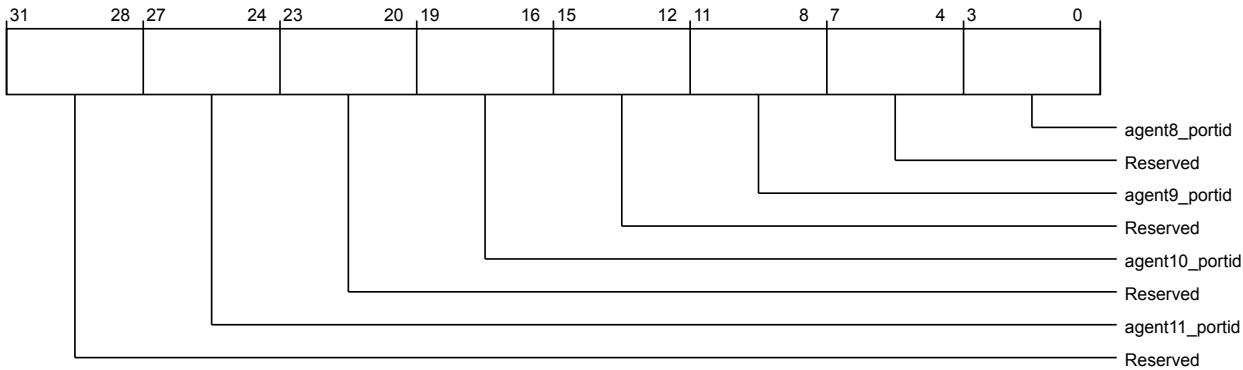


Figure 3-306 por_cxla_por_cxla_agentid_to_portid_reg1 (low)

The following table shows the por_cxla_agentid_to_portid_reg1 lower register bit assignments.

Table 3-326 por_cxla_por_cxla_agentid_to_portid_reg1 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent11_portid	Specifies the Port ID for Agent ID 11	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent10_portid	Specifies the Port ID for Agent ID 10	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent9_portid	Specifies the Port ID for Agent ID 9	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent8_portid	Specifies the Port ID for Agent ID 8	RW	4'h0

por_cxla_agentid_to_portid_reg2

Specifies the mapping of Agent ID to Port ID for Agent IDs 16 to 23.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

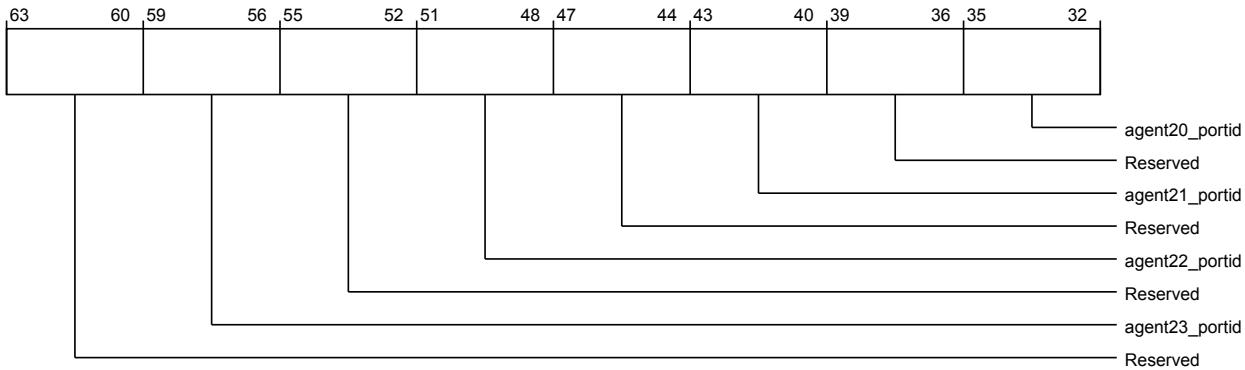


Figure 3-307 por_cxla_por_cxla_agentid_to_portid_reg2 (high)

The following table shows the por_cxla_agentid_to_portid_reg2 higher register bit assignments.

Table 3-327 por_cxla_por_cxla_agentid_to_portid_reg2 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent23_portid	Specifies the Port ID for Agent ID 23	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent22_portid	Specifies the Port ID for Agent ID 22	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent21_portid	Specifies the Port ID for Agent ID 21	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent20_portid	Specifies the Port ID for Agent ID 20	RW	4'h0

The following image shows the lower register bit assignments.

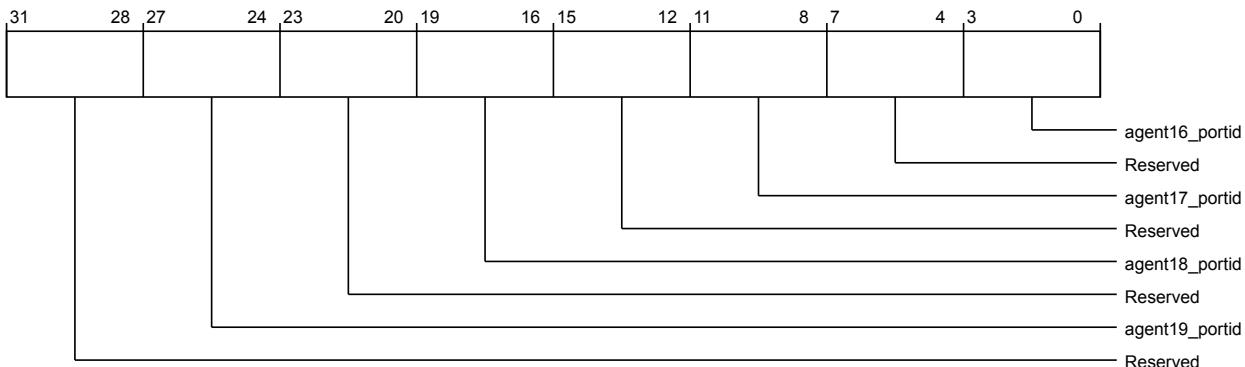


Figure 3-308 por_cxla_por_cxla_agentid_to_portid_reg2 (low)

The following table shows the por_cxla_agentid_to_portid_reg2 lower register bit assignments.

Table 3-328 por_cxla_por_cxla_agentid_to_portid_reg2 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent19_portid	Specifies the Port ID for Agent ID 19	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent18_portid	Specifies the Port ID for Agent ID 18	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent17_portid	Specifies the Port ID for Agent ID 17	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent16_portid	Specifies the Port ID for Agent ID 16	RW	4'h0

por_cxla_agentid_to_portid_reg3

Specifies the mapping of Agent ID to Port ID for Agent IDs 24 to 31.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	portcxla secure register groups override portid ctrl

override

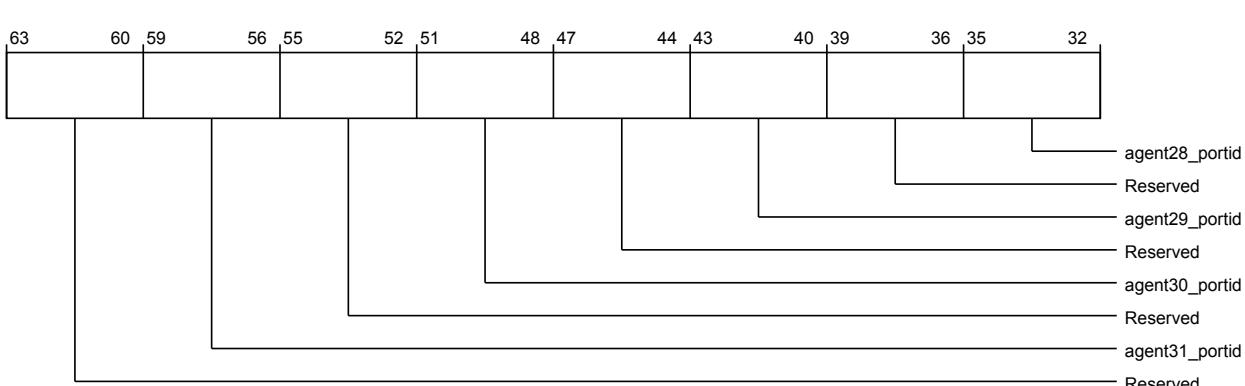


Figure 3-309 por cxla por cxla agentid to portid req3 (high)

The following table shows the portid to reg3 higher register bit assignments.

Table 3-329 por_cxla_por_cxla_agentid_to_portid_reg3 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent31_portid	Specifies the Port ID for Agent ID 31	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent30_portid	Specifies the Port ID for Agent ID 30	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent29_portid	Specifies the Port ID for Agent ID 29	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent28_portid	Specifies the Port ID for Agent ID 28	RW	4'h0

The following image shows the lower register bit assignments.

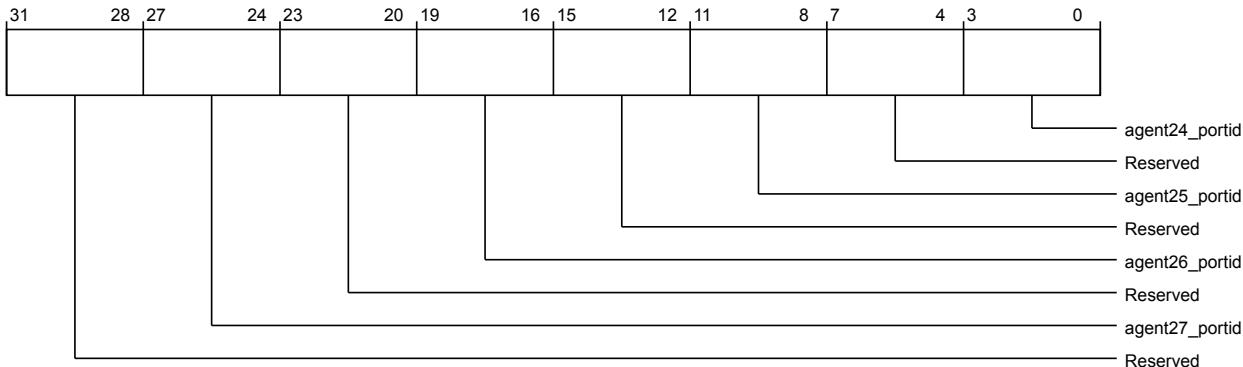


Figure 3-310 por_cxla_por_cxla_agentid_to_portid_reg3 (low)

The following table shows the por_cxla_agentid_to_portid_reg3 lower register bit assignments.

Table 3-330 por_cxla_por_cxla_agentid_to_portid_reg3 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent27_portid	Specifies the Port ID for Agent ID 27	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent26_portid	Specifies the Port ID for Agent ID 26	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent25_portid	Specifies the Port ID for Agent ID 25	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent24_portid	Specifies the Port ID for Agent ID 24	RW	4'h0

por_cxla_agentid_to_portid_reg4

Specifies the mapping of Agent ID to Port ID for Agent IDs 32 to 39.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD50
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

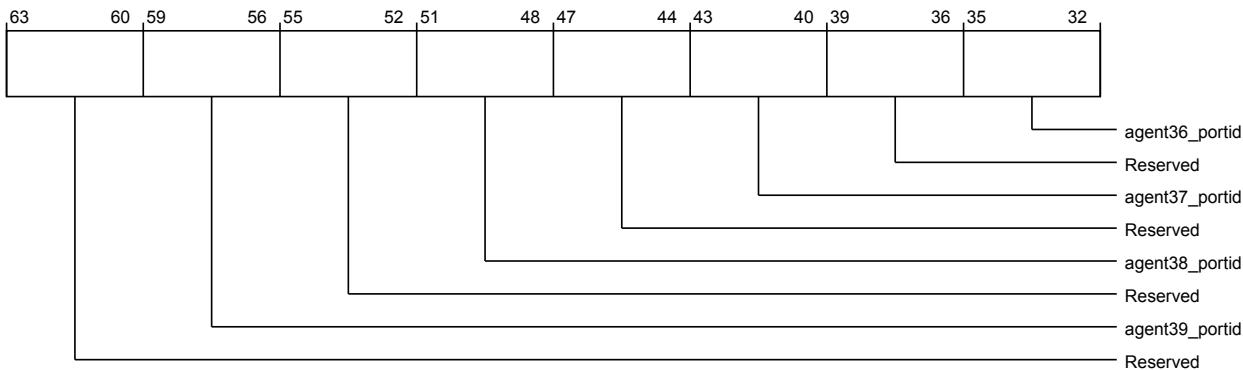


Figure 3-311 por_cxla_por_cxla_agentid_to_portid_reg4 (high)

The following table shows the por_cxla_agentid_to_portid_reg4 higher register bit assignments.

Table 3-331 por_cxla_por_cxla_agentid_to_portid_reg4 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent39_portid	Specifies the Port ID for Agent ID 39	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent38_portid	Specifies the Port ID for Agent ID 38	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent37_portid	Specifies the Port ID for Agent ID 37	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent36_portid	Specifies the Port ID for Agent ID 36	RW	4'h0

The following image shows the lower register bit assignments.

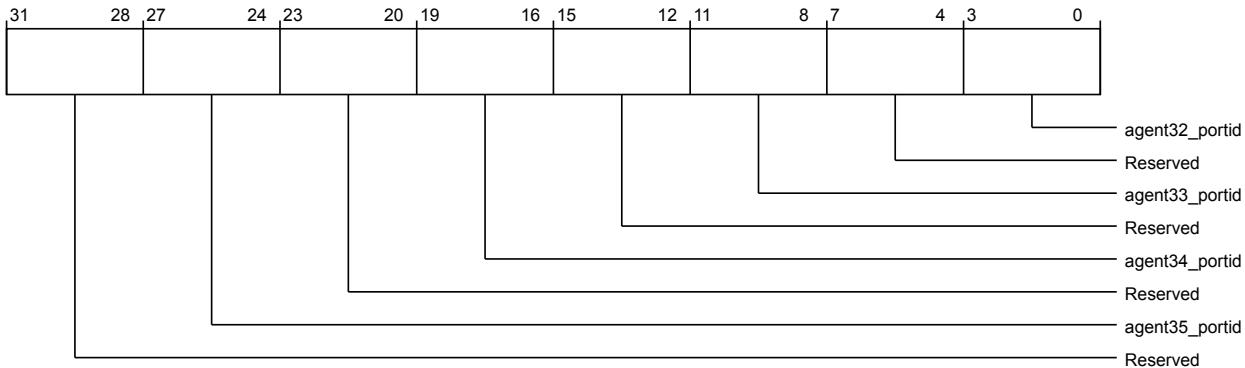


Figure 3-312 por_cxla_por_cxla_agentid_to_portid_reg4 (low)

The following table shows the por_cxla_agentid to portid_reg4 lower register bit assignments.

Table 3-332 por_cxla_por_cxla_agentid_to_portid_reg4 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent35_portid	Specifies the Port ID for Agent ID 35	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent34_portid	Specifies the Port ID for Agent ID 34	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent33_portid	Specifies the Port ID for Agent ID 33	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent32_portid	Specifies the Port ID for Agent ID 32	RW	4'h0

por_cxla_agentid_to_portid_reg5

Specifies the mapping of Agent ID to Port ID for Agent IDs 40 to 47.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_cxla_secure_register_groups_override.portid_ctl

The following is a code example which illustrates this issue:

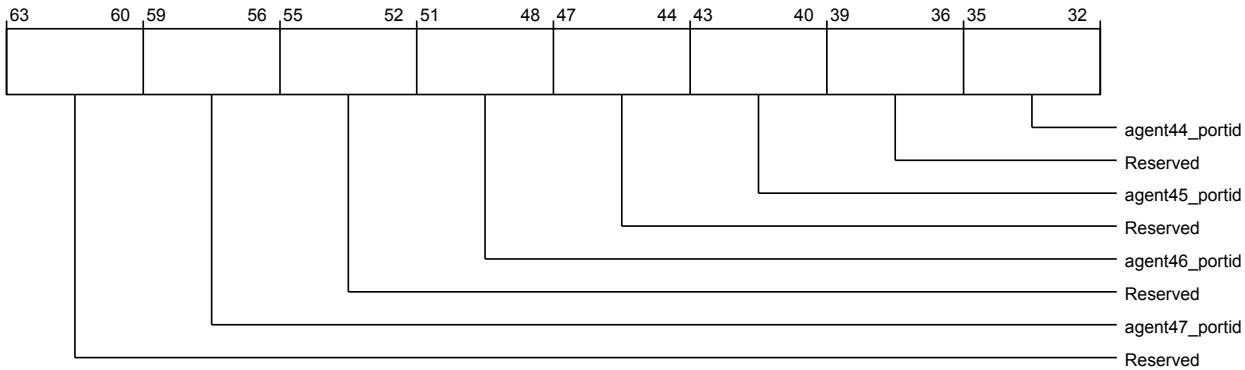


Figure 3-313 por_cxla_por_cxla_agentid_to_portid_reg5 (high)

The following table shows the por_cxla_agentid_to_portid_reg5 higher register bit assignments.

Table 3-333 por_cxla_por_cxla_agentid_to_portid_reg5 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent47_portid	Specifies the Port ID for Agent ID 47	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent46_portid	Specifies the Port ID for Agent ID 46	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent45_portid	Specifies the Port ID for Agent ID 45	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent44_portid	Specifies the Port ID for Agent ID 44	RW	4'h0

The following image shows the lower register bit assignments.

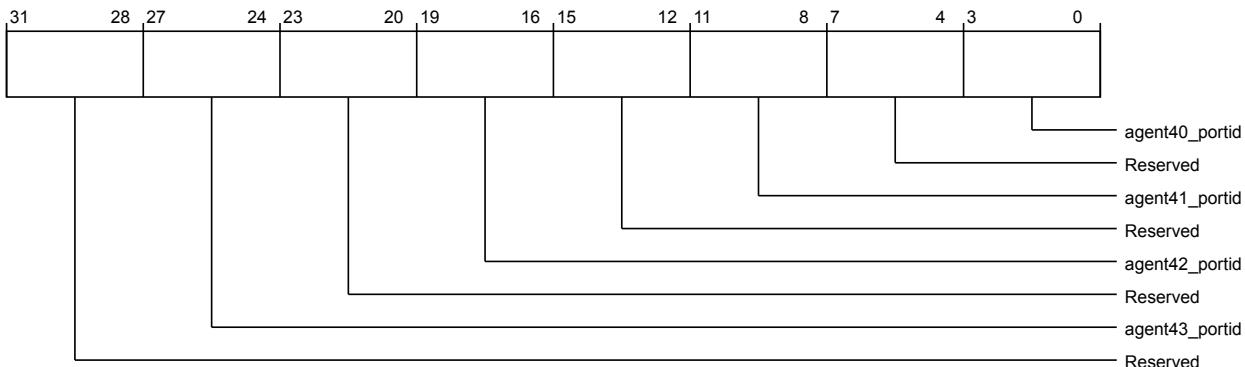


Figure 3-314 por_cxla_por_cxla_agentid_to_portid_reg5 (low)

The following table shows the por_cxla_agentid_to_portid_reg5 lower register bit assignments.

Table 3-334 por_cxla_por_cxla_agentid_to_portid_reg5 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent43_portid	Specifies the Port ID for Agent ID 43	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent42_portid	Specifies the Port ID for Agent ID 42	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent41_portid	Specifies the Port ID for Agent ID 41	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent40_portid	Specifies the Port ID for Agent ID 40	RW	4'h0

por_cxla_agentid_to_portid_reg6

Specifies the mapping of Agent ID to Port ID for Agent IDs 48 to 55.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

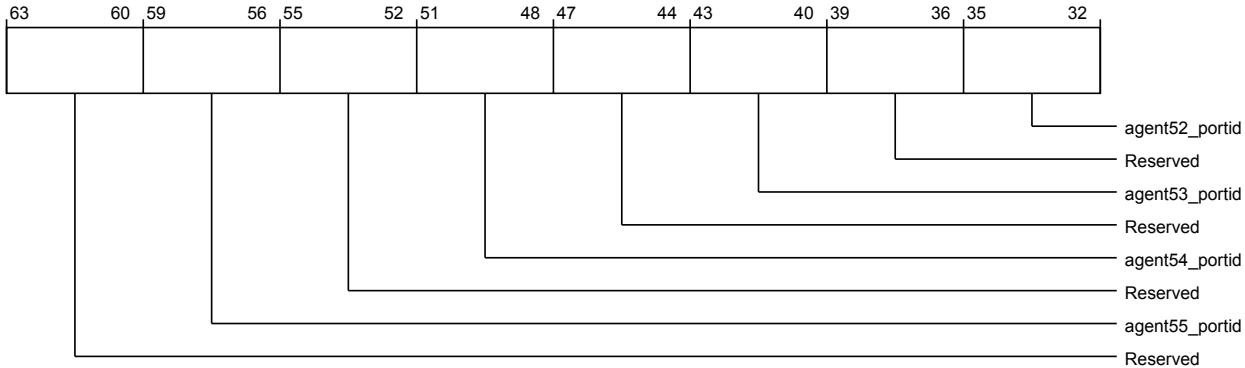


Figure 3-315 por_cxla_por_cxla_agentid_to_portid_reg6 (high)

The following table shows the por_cxla_agentid_to_portid_reg6 higher register bit assignments.

Table 3-335 por_cxla_por_cxla_agentid_to_portid_reg6 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent55_portid	Specifies the Port ID for Agent ID 55	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent54_portid	Specifies the Port ID for Agent ID 54	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent53_portid	Specifies the Port ID for Agent ID 53	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent52_portid	Specifies the Port ID for Agent ID 52	RW	4'h0

The following image shows the lower register bit assignments.

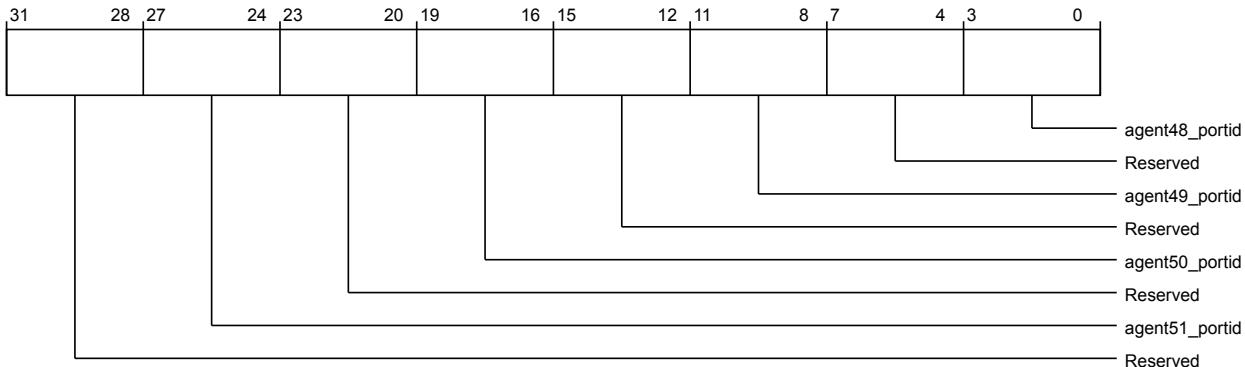


Figure 3-316 por_cxla_por_cxla_agentid_to_portid_reg6 (low)

The following table shows the por_cxla_agentid_to_portid_reg6 lower register bit assignments.

Table 3-336 por_cxla_por_cxla_agentid_to_portid_reg6 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent51_portid	Specifies the Port ID for Agent ID 51	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent50_portid	Specifies the Port ID for Agent ID 50	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent49_portid	Specifies the Port ID for Agent ID 49	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent48_portid	Specifies the Port ID for Agent ID 48	RW	4'h0

por_cxla_agentid_to_portid_reg7

Specifies the mapping of Agent ID to Port ID for Agent IDs 56 to 63.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD68
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

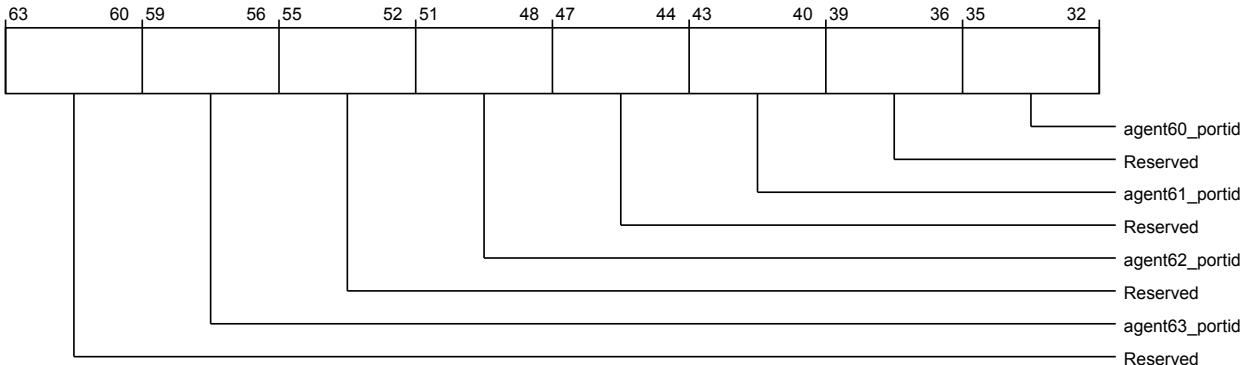


Figure 3-317 por_cxla_por_cxla_agentid_to_portid_reg7 (high)

The following table shows the port pin assignments for the higher register bit assignments.

Table 3-337 por_cxla_por_cxla_agentid_to_portid_reg7 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent63_portid	Specifies the Port ID for Agent ID 63	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent62_portid	Specifies the Port ID for Agent ID 62	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent61_portid	Specifies the Port ID for Agent ID 61	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent60_portid	Specifies the Port ID for Agent ID 60	RW	4'h0

The following image shows the lower register bit assignments.

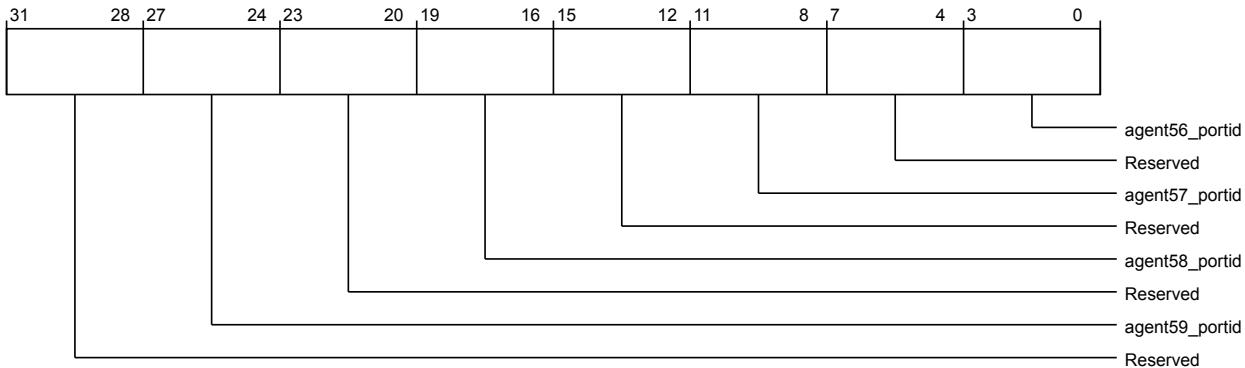


Figure 3-318 por_cxla_por_cxla_agentid_to_portid_reg7 (low)

The following table shows the por_cxla_agentid to portid_reg7 lower register bit assignments.

Table 3-338 por_cxla_por_cxla_agentid_to_portid_reg7 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent59_portid	Specifies the Port ID for Agent ID 59	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent58_portid	Specifies the Port ID for Agent ID 58	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent57_portid	Specifies the Port ID for Agent ID 57	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent56_portid	Specifies the Port ID for Agent ID 56	RW	4'h0

por_cxla_agentid_to_portid_val

Specifies which Agent ID to Port ID mappings are valid.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_cxla_secure_register_groups_override.portid_ctl

The following sections describe the high-level architecture of the system.

63	valid	32
----	-------	----

Figure 3-319 por_cxla_por_cxla_agentid_to_portid_val (high)

The following table shows the por_cxla_agentid_to_portid_val higher register bit assignments.

Table 3-339 por_cxla_por_cxla_agentid_to_portid_val (high)

Bits	Field name	Description	Type	Reset
63:32	valid	Specifies whether the Port ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

The following image shows the lower register bit assignments.

31	valid	0
----	-------	---

Figure 3-320 por_cxla_por_cxla_agentid_to_portid_val (low)

The following table shows the por_cxla_agentid_to_portid_val lower register bit assignments.

Table 3-340 por_cxla_por_cxla_agentid_to_portid_val (low)

Bits	Field name	Description	Type	Reset
31:0	valid	Specifies whether the Port ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

por_cxla_portfwd_ctl

Functions as the Port-to-Port forwarding control register. Works with por_cxla_portfwd_status.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD78

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63	Reserved	48	47	32
			port_fwd_req	

Figure 3-321 por_cxla_por_cxla_portfwd_ctl (high)

The following table shows the por_cxla_portfwd_ctl higher register bit assignments.

Table 3-341 por_cxla_por_cxla_portfwd_ctl (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	port_fwd_req	Bit vector, where each bit represents logical PortID of a Port (CXG) present on CMN. Each bit is used to control the communication channel with the corresponding port. 1'b0: Port-to-Port forwarding channel de-activate request 1'b1: Port-to-Port forwarding channel activate request	RW	16'b0

The following image shows the lower register bit assignments.

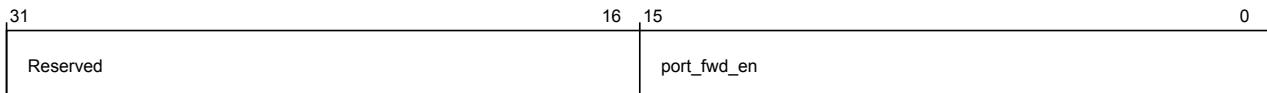


Figure 3-322 por_cxla_por_cxla_portfwd_ctl (low)

The following table shows the por_cxla_portfwd_ctl lower register bit assignments.

Table 3-342 por_cxla_por_cxla_portfwd_ctl (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	port_fwd_en	Bit vector, where each bit represents logical PortID of a Port (CXG) present on CMN. Each bit when set, enables Port-to-Port forwarding with the corresponding port. 1'b0: Port-to-Port forwarding is disabled 1'b1: Port-to-Port forwarding is enabled	RW	16'b0

por_cxla_portfwd_status

Functions as the Port-to-Port forwarding status register. Works with por_cxla_portfwd_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hD80

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-323 por_cxla_por_cxla_portfwd_status (high)

The following table shows the por_cxla_portfwd_status higher register bit assignments.

Table 3-343 por_cxla_por_cxla_portfwd_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31

16 15

0

Reserved

port_fwd_ack

Figure 3-324 por_cxla_por_cxla_portfwd_status (low)

The following table shows the por_cxla_portfwd_status lower register bit assignments.

Table 3-344 por_cxla_por_cxla_portfwd_status (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	port_fwd_ack	Bit vector, where each bit represents logical PortID of a Port (CXG) present on CMN. Each bit represents the status of the port-to-port control request sent to the corresponding port. 1'b0: Port-to-Port forwarding channel is de-active. 1'b1: Port-to-Port forwarding channel is active	RO	16'b0

por_cxla_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

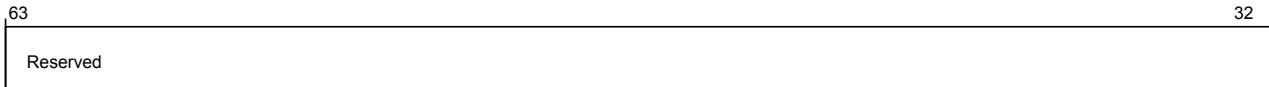


Figure 3-325 por_cxla_por_cxla_pmu_event_sel (high)

The following table shows the por_cxla_pmu_event_sel higher register bit assignments.

Table 3-345 por_cxla_por_cxla_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

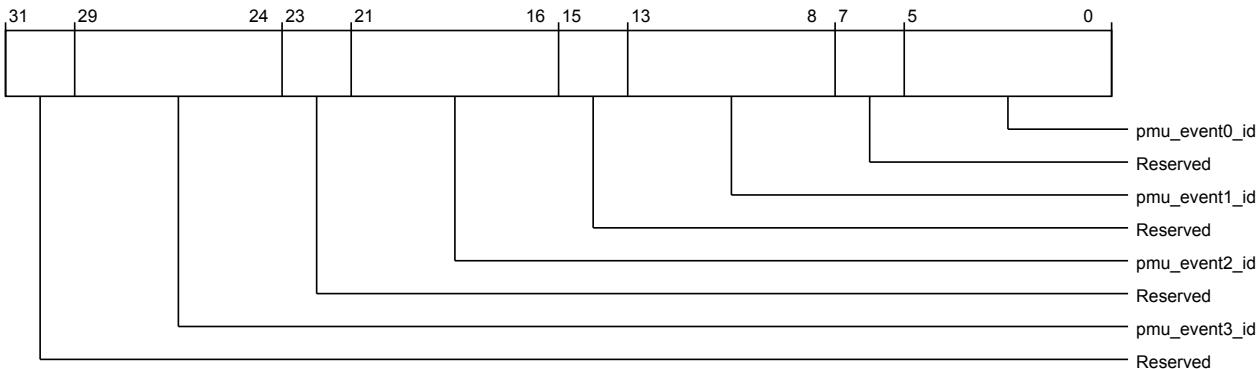


Figure 3-326 por_cxla_por_cxla_pmu_event_sel (low)

The following table shows the por_cxla_pmu_event_sel lower register bit assignments.

Table 3-346 por_cxla_por_cxla_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	CXLA PMU Event 3 ID; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	CXLA PMU Event 2 ID; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	CXLA PMU Event 1 ID; see pmu_event0_id for encodings	RW	6'b0

Table 3-346 por_cxla_por_cxla_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-

Table 3-346 por_cxla_por_cxla_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
5:0	pmu_event0_id	CXLA PMU Event 0 ID 6'h00: No event 6'h01: Average RX TLP count for Link 0 6'h02: Average RX TLP count for Link 1 6'h03: Average RX TLP count for Link 2 6'h04: Average TX TLP count for Link 0 6'h05: Average TX TLP count for Link 1 6'h06: Average TX TLP count for Link 2 6'h07: RX CXS for Link 0 6'h08: RX CXS for Link 1 6'h09: RX CXS for Link 2 6'h0A: TX CXS for Link 0 6'h0B: TX CXS for Link 1 6'h0C: TX CXS for Link 2 6'h0D: Average RX TLP size in DWs 6'h0E: Average TX TLP size in DWs 6'h0F: Average RX TLP size in CCIX messages 6'h10: Average TX TLP size in CCIX messages 6'h11: Average size of RX CXS in DWs within a beat 6'h12: Average size of TX CXS in DWs within a beat 6'h13: TX CXS link credit backpressure 6'h14: RX TLP buffer full and backpressured 6'h15: TX TLP buffer full and backpressured 6'h16: Reserved 6'h17: Average latency to form a TX TLP 6'h18: TX Request Chain 6'h19: RX RSPDAT (Memory Response with Data) CGL Buffer backpressured 6'h1A: RX SNP DAT (Snoop Response with Data) CGL Buffer backpressured 6'h1B: RX REQ DAT (Memory Request) CGL Buffer backpressured 6'h1C: RX REQRSP (Memory Response without Data) CGL Buffer backpressured 6'h1D: RX SNPRSP (Snoop Response without Data) CGL Buffer backpressured 6'h1E: RX SNP (Snoop Request) CGL Buffer backpressured 6'h1F: RX UCMISC (Misc) CGL Buffer backpressured 6'h20: RX Number of Port forwarded CXS beats 6'h21: TX Number of Port forwarded CXS beats 6'h22: RX Number of port forwarded message stalls due to static credits 6'h23: RX Number of port forwarded message stalls due to dynamic credits	RW	6'b0

por_cxla_pmu_config

Configures the CXLA PMU.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2210

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-327 por_cxla_por_cxla_pmu_config (high)

The following table shows the por_cxla_pmu_config higher register bit assignments.

Table 3-347 por_cxla_por_cxla_pmu_config (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

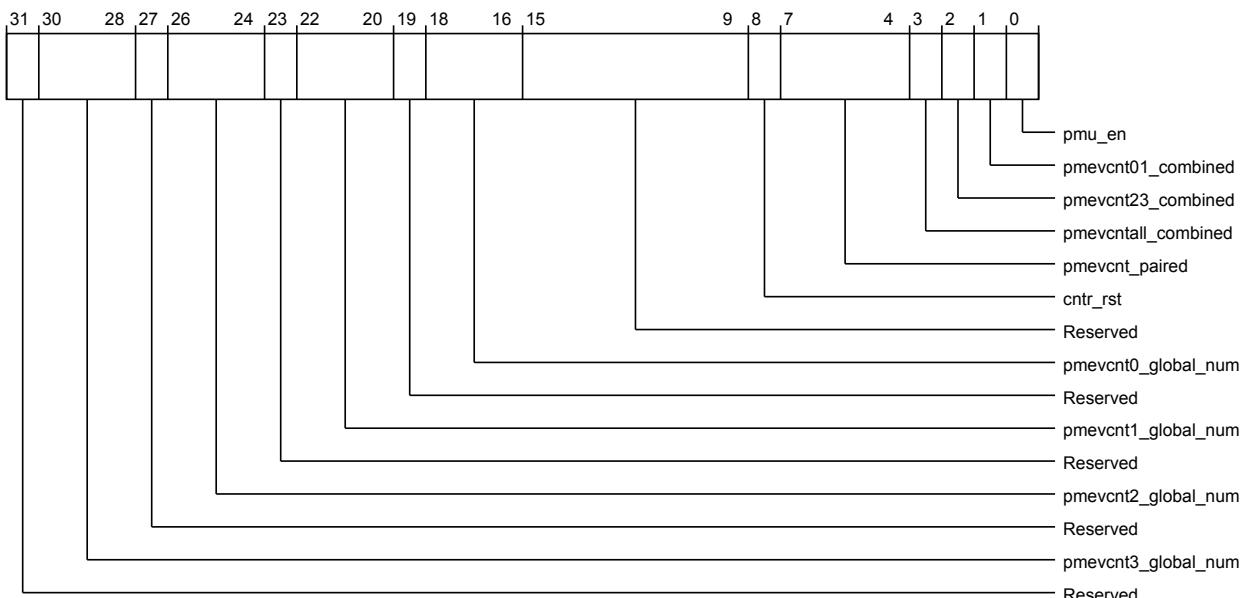


Figure 3-328 por_cxla_por_cxla_pmu_config (low)

The following table shows the por_cxla_pmu_config lower register bit assignments.

Table 3-348 por_cxla_por_cxla_pmu_config (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	pmevcnt3_global_num	Global counter to pair with PMU counter 3; see pmevcnt0_global_num for encodings	RW	3'b0
27	Reserved	Reserved	RO	-
26:24	pmevcnt2_global_num	Global counter to pair with PMU counter 2; see pmevcnt0_global_num for encodings	RW	3'b0
23	Reserved	Reserved	RO	-
22:20	pmevcnt1_global_num	Global counter to pair with PMU counter 1; see pmevcnt0_global_num for encodings	RW	3'b0
19	Reserved	Reserved	RO	-
18:16	pmevcnt0_global_num	Global counter to pair with PMU counter 0 3'b000: Global PMU event counter A 3'b001: Global PMU event counter B 3'b010: Global PMU event counter C 3'b011: Global PMU event counter D 3'b100: Global PMU event counter E 3'b101: Global PMU event counter F 3'b110: Global PMU event counter G 3'b111: Global PMU event counter H	RW	3'b0
15:9	Reserved	Reserved	RO	-
8	cntr_rst	Enables clearing of live counters upon assertion of snapshot	RW	1'b0
7:4	pmevcnt_paired	PMU local counter paired with global counter	RW	4'b0
3	pmevcntall_combined	Enables combination of all PMU counters (0, 1, 2, 3) NOTE: When set, pmevcnt01_combined and pmevcnt23_combined have no effect.	RW	1'b0
2	pmevcnt23_combined	Enables combination of PMU counters 2 and 3	RW	1'b0
1	pmevcnt01_combined	Enables combination of PMU counters 0 and 1	RW	1'b0
0	pmu_en	CXLA PMU enable NOTE: All other fields in this register are valid only if this bit is set.	RW	1'b0

por_cxla_pmevcnt

Contains all PMU event counters (0, 1, 2, 3).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2220

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	48	47	32
pmevcnt3		pmevcnt2	

Figure 3-329 por_cxla_por_cxla_pmevcnt (high)

The following table shows the por_cxla_pmevcnt higher register bit assignments.

Table 3-349 por_cxla_por_cxla_pmevcnt (high)

Bits	Field name	Description	Type	Reset
63:48	pmevcnt3	PMU event counter 3	RW	16'h0000
47:32	pmevcnt2	PMU event counter 2	RW	16'h0000

The following image shows the lower register bit assignments.

31	16	15	0
pmevcnt1		pmevcnt0	

Figure 3-330 por_cxla_por_cxla_pmevcnt (low)

The following table shows the por_cxla_pmevcnt lower register bit assignments.

Table 3-350 por_cxla_por_cxla_pmevcnt (low)

Bits	Field name	Description	Type	Reset
31:16	pmevcnt1	PMU event counter 1	RW	16'h0000
15:0	pmevcnt0	PMU event counter 0	RW	16'h0000

por_cxla_pmevcntsr

Functions as the PMU event counter shadow register for all counters (0, 1, 2, 3).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2240

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	48 47	32
pmevcntsr3	pmevcntsr2	

Figure 3-331 por_cxla_por_cxla_pmevcntsr (high)

The following table shows the por_cxla_pmevcntsr higher register bit assignments.

Table 3-351 por_cxla_por_cxla_pmevcntsr (high)

Bits	Field name	Description	Type	Reset
63:48	pmevcntsr3	PMU event counter 3 shadow register	RW	16'h0000
47:32	pmevcntsr2	PMU event counter 2 shadow register	RW	16'h0000

The following image shows the lower register bit assignments.

31	16 15	0
pmevcntsr1	pmevcntsr0	

Figure 3-332 por_cxla_por_cxla_pmevcntsr (low)

The following table shows the por_cxla_pmevcntsr lower register bit assignments.

Table 3-352 por_cxla_por_cxla_pmevcntsr (low)

Bits	Field name	Description	Type	Reset
31:16	pmevcntsr1	PMU event counter 1 shadow register	RW	16'h0000
15:0	pmevcntsr0	PMU event counter 0 shadow register	RW	16'h0000

3.3.7 DN register descriptions

This section lists the DN registers.

por_dn_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	48 47	32
Reserved	logical_id	

Figure 3-333 por_dn_por_dn_node_info (high)

The following table shows the por_dn_node_info higher register bit assignments.

Table 3-353 por_dn_por_dn_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

31	16 15	0
node_id	node_type	

Figure 3-334 por_dn_por_dn_node_info (low)

The following table shows the por_dn_node_info lower register bit assignments.

Table 3-354 por_dn_por_dn_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0001

por_dn_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

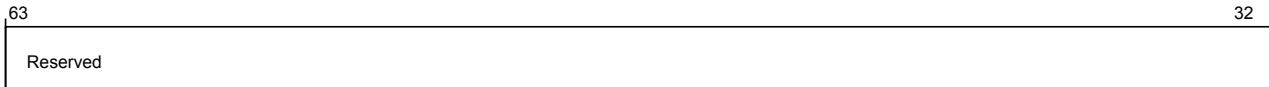


Figure 3-335 por_dn_por_dn_child_info (high)

The following table shows the por_dn_child_info higher register bit assignments.

Table 3-355 por_dn_por_dn_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

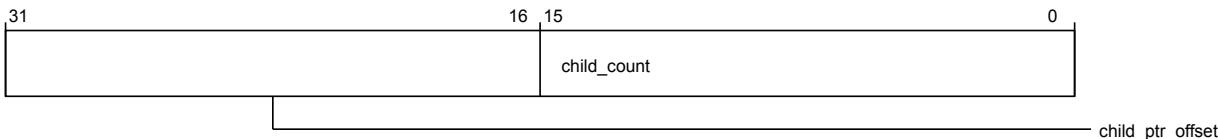


Figure 3-336 por_dn_por_dn_child_info (low)

The following table shows the por_dn_child_info lower register bit assignments.

Table 3-356 por_dn_por_dn_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'h0

por_dn_build_info

Contains the configuration parameter values. Indicates the specific DN configuration.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset 64'b1

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-337 por_dn_por_dn_build_info (high)

The following table shows the por_dn_build_info higher register bit assignments.

Table 3-357 por_dn_por_dn_build_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

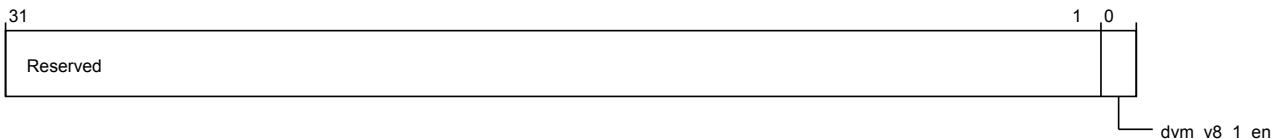


Figure 3-338 por_dn_por_dn_build_info (low)

The following table shows the por_dn_build_info lower register bit assignments.

Table 3-358 por_dn_por_dn_build_info (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	dvm_v8_1_en	Determines whether all nodes receiving DVM snoops support DVM v8.1 operations; must be set to 0 if not supported by all nodes, therefore allowing the node to perform demotion before sending out the DVM snoop	RO	1'b1

por_dn_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

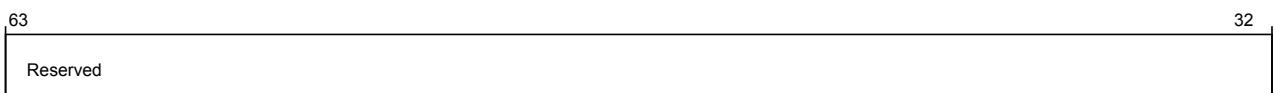


Figure 3-339 por_dn_por_dn_secure_register_groups_override (high)

The following table shows the por_dn_secure_register_groups_override higher register bit assignments.

Table 3-359 por_dn_por_dn_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

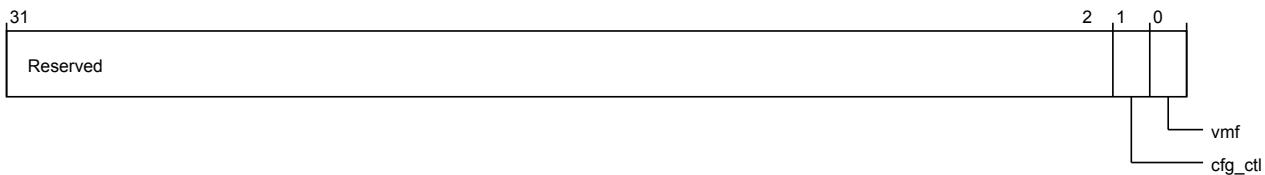


Figure 3-340 por_dn_por_dn_secure_register_groups_override (low)

The following table shows the por_dn_secure_register_groups_override lower register bit assignments.

Table 3-360 por_dn_por_dn_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1	cfg_ctl	Allows non-secure access to secure configuration control register (por_dn_cfg_ctl)	RW	1'b0
0	vmf	Allows non-secure access to secure VMF registers	RW	1'b0

por_dn_cfg_ctl

Functions as the configuration control register for DVM Node.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b010000
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_dn_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

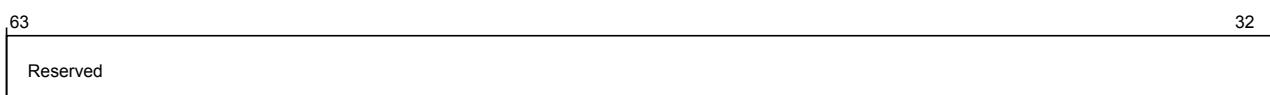


Figure 3-341 por_dn_por_dn_cfg_ctl (high)

The following table shows the por_dn_cfg_ctl higher register bit assignments.

Table 3-361 por_dn_por_dn_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

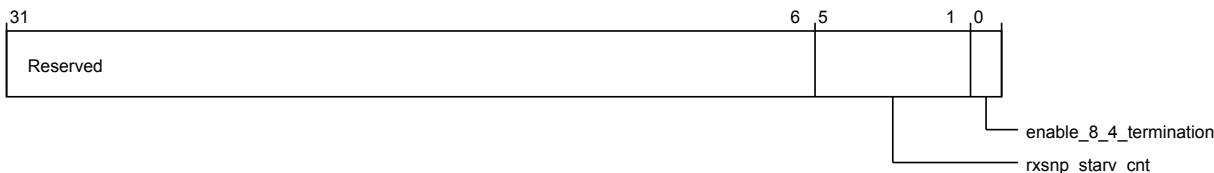


Figure 3-342 por_dn_por_dn_cfg_ctl (low)

The following table shows the por_dn_cfg_ctl lower register bit assignments.

Table 3-362 por_dn_por_dn_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5:1	rxsnr_starv_cnt	Number of cycles RXSNR lost to RXREQ for RCB alloc.	RW	5'h8
0	enable_8_4_termination	Enables termination of 8.4 DVMOps in DN.	RW	1'b0

por_dn_aux_ctl

Functions as the auxiliary control register for DN.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

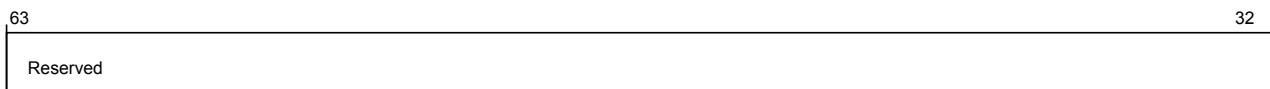


Figure 3-343 por_dn_por_dn_aux_ctl (high)

The following table shows the por_dn_aux_ctl higher register bit assignments.

Table 3-363 por_dn_por_dn_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

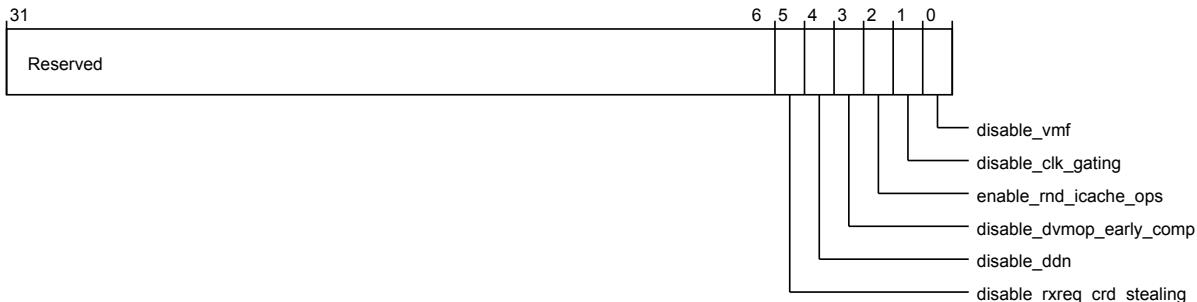


Figure 3-344 por_dn_por_dn_aux_ctl (low)

The following table shows the por_dn_aux_ctl lower register bit assignments.

Table 3-364 por_dn_por_dn_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5	disable_rxreq_crd_stealing	Disables credit stealing from RXREQ LinkLayer when RXSNP is starved for RCB alloc.	RW	1'b0
4	disable_ddn	Disables Distributed DN functionality- Snoops all RNs and CML nodes in the mesh and disables snooping other DNs. Must program all RNSAMs to target HND for DVMs and then set this to 1 in HND.	RW	1'b0
3	disable_dvmop_early_comp	Disables Early Comp (CompDBID) for DVMOps	RW	1'b1
2	enable_rnd_icache_ops	Filters out BPI and VICI/PICI Snps to RNDs when set	RW	Configuration dependent
1	disable_clk_gating	Disables autonomous clock gating when set	RW	1'b0
0	disable_vmf	This bit is currently not supported. Software must not program this bit.	RW	Configuration dependent

por_dn_vmf_0-15_ctrl

This register repeats 15 times. It parametrized by the vmf_id from 0 to 15. Functions as the control register for VMID-based DVM snoop filtering. NOTE: This register has no effect when por_dn_aux_ctl.disable_vmf is set to 1.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset	16'hC00 + #{56*[0, 1, 2, .., 14, 15]}
Register reset	64'b1111111111111111000000000000000000000000
Usage constraints	Only accessible by secure accesses.
Secure group override	por_dn_secure_register_groups_override.vmf

The following image shows the higher register bit assignments.

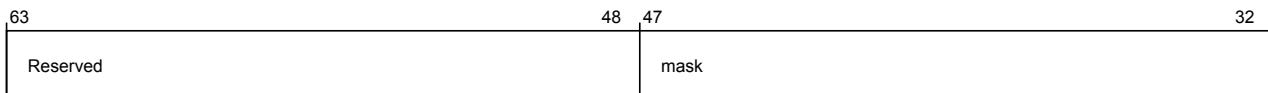


Figure 3-345 por_dn_por_dn_vmf_0-15_ctrl (high)

The following table shows the por_dn_vmf_0-15_ctrl higher register bit assignments.

Table 3-365 por_dn_por_dn_vmf_0-15_ctrl (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	mask	VMID mask; enables mapping of multiple VMID values to a single register NOTE: Logically, the AND operator is performed on the mask and por_dn_vmf#{vmf_id}_ctrl.vmid. Then, the AND operator is performed on the mask and the incoming request's VMID. The two results are then compared, and filtering is applied to the incoming request if the masked VMIDs match.	RW	16'hffff

The following image shows the lower register bit assignments.

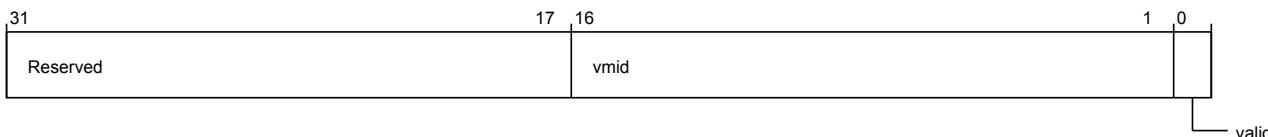


Figure 3-346 por_dn_por_dn_vmf_0-15_ctrl (low)

The following table shows the por_dn_vmf_0-15_ctrl lower register bit assignments.

Table 3-366 por_dn_por_dn_vmf_0-15_ctrl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16:1	vmid	VMID value NOTE: The incoming request's VMID is only compared with this VMID value if the request's VMID valid bit is set. If the request's VMID is valid and the two VMIDs match, filtering is applied to the incoming request.	RW	16'h0000
0	valid	Register valid 1'b1: Register is enabled 1'b0: Register is not enabled	RW	1'b0

por_dn_vmf_0-15_rnf0

This register repeats 15 times. It parametrized by the vmf_id from 0 to 15. Contains the logical RN-F bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid. Used for VMID-based DVM snoop filtering.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + #{56*[0, 1, 2, .., 14, 15]+8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_dn_secure_register_groups_override.vmf

The following image shows the higher register bit assignments.

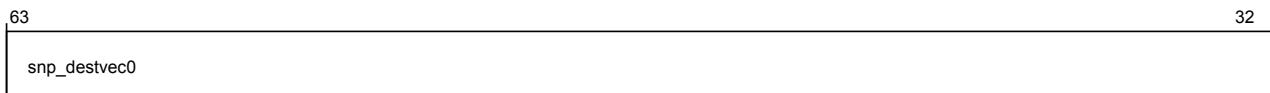


Figure 3-347 por_dn_por_dn_vmf_0-15_rnf0 (high)

The following table shows the por_dn_vmf_0-15_rnf0 higher register bit assignments.

Table 3-367 por_dn_por_dn_vmf_0-15_rnf0 (high)

Bits	Field name	Description	Type	Reset
63:32	snp_destvec0	RN-F bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

The following image shows the lower register bit assignments.

31	snp_destvec0	0
----	--------------	---

Figure 3-348 por_dn_por_dn_vmf_0-15_rnf0 (low)

The following table shows the por_dn_vmf_0-15_rnf0 lower register bit assignments.

Table 3-368 por_dn_por_dn_vmf_0-15_rnf0 (low)

Bits	Field name	Description	Type	Reset
31:0	snp_destvec0	RN-F bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

por_dn_vmf_0-15_rnf1

This register repeats 15 times. It parametrized by the vmf_id from 0 to 15. Contains the logical RN-F bit vector 127:64 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid. Used for VMID-based DVM snoop filtering.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + #{56*[0, 1, 2, .., 14, 15]+16}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_dn_secure_register_groups_override.vmf

The following image shows the higher register bit assignments.

63	snp_destvec1	32
----	--------------	----

Figure 3-349 por_dn_por_dn_vmf_0-15_rnf1 (high)

The following table shows the por_dn_vmf_0-15_rnf1 higher register bit assignments.

Table 3-369 por_dn_por_dn_vmf_0-15_rnf1 (high)

Bits	Field name	Description	Type	Reset
63:32	snp_destvec1	RN-F bit vector 127:64 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

The following image shows the lower register bit assignments.

31	snp_destvec1	0
----	--------------	---

Figure 3-350 por_dn_por_dn_vmf_0-15_rnf1 (low)

The following table shows the por_dn_vmf_0-15_rnf1 lower register bit assignments.

Table 3-370 por_dn_por_dn_vmf_0-15_rnf1 (low)

Bits	Field name	Description	Type	Reset
31:0	snp_destvec1	RN-F bit vector 127:64 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

por_dn_vmf_0-15_rnf2

This register repeats 15 times. It parametrized by the vmf_id from 0 to 15. Contains the logical RN-F bit vector 191:128 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid. Used for VMID-based DVM snoop filtering.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + #{56*[0, 1, 2, .., 14, 15]+24}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_dn_secure_register_groups_override.vmf

The following image shows the higher register bit assignments.

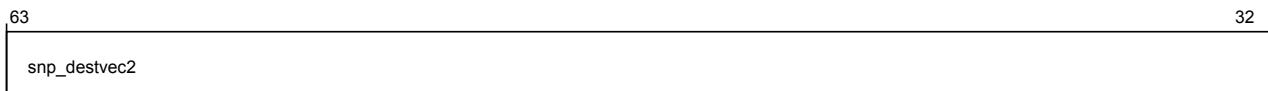


Figure 3-351 por_dn_por_dn_vmf_0-15_rnf2 (high)

The following table shows the por_dn_vmf_0-15_rnf2 higher register bit assignments.

Table 3-371 por_dn_por_dn_vmf_0-15_rnf2 (high)

Bits	Field name	Description	Type	Reset
63:32	snp_destvec2	RN-F bit vector 191:128 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

The following image shows the lower register bit assignments.

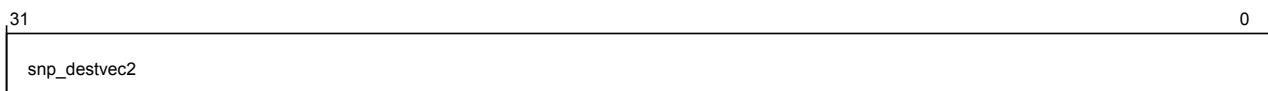


Figure 3-352 por_dn_por_dn_vmf_0-15_rnf2 (low)

The following table shows the por_dn_vmf_0-15_rnf2 lower register bit assignments.

Table 3-372 por_dn_por_dn_vmf_0-15_rnf2 (low)

Bits	Field name	Description	Type	Reset
31:0	snp_destvec2	RN-F bit vector 191:128 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

por_dn_vmf_0-15_rnf3

This register repeats 15 times. It parametrized by the vmf_id from 0 to 15. Contains the logical RN-F bit vector 255:192 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid. Used for VMID-based DVM snoop filtering.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + #{56*[0, 1, 2, .., 14, 15]+32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_dn_secure_register_groups_override.vmf

The following image shows the higher register bit assignments.



Figure 3-353 por_dn_por_dn_vmf_0-15_rnf3 (high)

The following table shows the por_dn_vmf_0-15_rnf3 higher register bit assignments.

Table 3-373 por_dn_por_dn_vmf_0-15_rnf3 (high)

Bits	Field name	Description	Type	Reset
63:32	snp_destvec3	RN-F bit vector 255:192 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

The following image shows the lower register bit assignments.

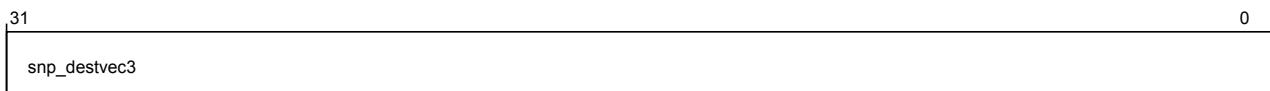


Figure 3-354 por_dn_por_dn_vmf_0-15_rnf3 (low)

The following table shows the por_dn_vmf_0-15_rnf3 lower register bit assignments.

Table 3-374 por_dn_por_dn_vmf_0-15_rnf3 (low)

Bits	Field name	Description	Type	Reset
31:0	snp_destvec3	RN-F bit vector 255:192 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

por_dn_vmf_0-15_rnd

This register repeats 15 times. It parametrized by the vmf_id from 0 to 15. Contains the logical RN-D bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid. Used for VMID-based DVM snoop filtering.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + #{56*[0, 1, 2, .., 14, 15]}+40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_dn_secure_register_groups_override.vmf

The following image shows the higher register bit assignments.

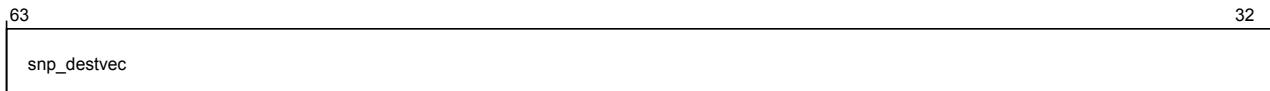


Figure 3-355 por_dn_por_dn_vmf_0-15_rnd (high)

The following table shows the por_dn_vmf_0-15_rnd higher register bit assignments.

Table 3-375 por_dn_por_dn_vmf_0-15_rnd (high)

Bits	Field name	Description	Type	Reset
63:32	snp_destvec	RN-D bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

The following image shows the lower register bit assignments.

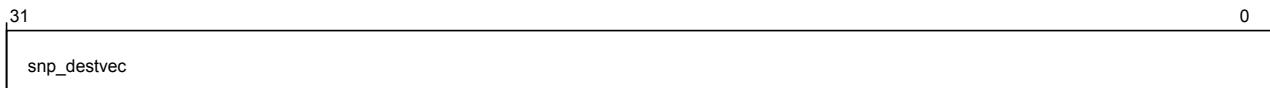


Figure 3-356 por_dn_por_dn_vmf_0-15_rnd (low)

The following table shows the por_dn_vmf_0-15_rnd lower register bit assignments.

Table 3-376 por_dn_por_dn_vmf_0-15_rnd (low)

Bits	Field name	Description	Type	Reset
31:0	snp_destvec	RN-D bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

por_dn_vmf_0-15_cxra

This register repeats 15 times. It parametrized by the vmf_id from 0 to 15. Contains the logical CXRA bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid. Used for VMID-based DVM snoop filtering. NOTE: Not applicable in a single-chip CMN-600 system. Does not have any effect.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + #{56*[0, 1, 2, .., 14, 15]+48}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_dn_secure_register_groups_override.vmf

The following image shows the higher register bit assignments.

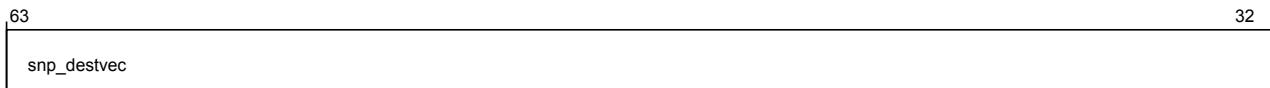


Figure 3-357 por_dn_por_dn_vmf_0-15_cxra (high)

The following table shows the por_dn_vmf_0-15_cxra higher register bit assignments.

Table 3-377 por_dn_por_dn_vmf_0-15_cxra (high)

Bits	Field name	Description	Type	Reset
63:32	snp_destvec	CXRA bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

The following image shows the lower register bit assignments.

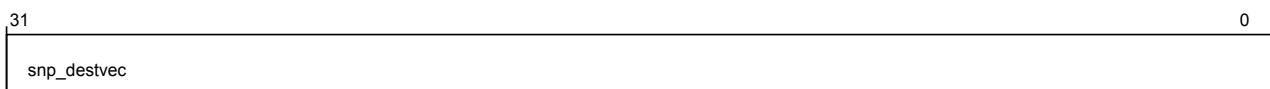


Figure 3-358 por_dn_por_dn_vmf_0-15_cxra (low)

The following table shows the por_dn_vmf_0-15_cxra lower register bit assignments.

Table 3-378 por_dn_por_dn_vmf_0-15_cxra (low)

Bits	Field name	Description	Type	Reset
31:0	snp_destvec	CXRA bit vector 63:0 corresponding to por_dn_vmf#{vmf_id}_ctrl.vmid	RW	64'b0

por_dn_domain_rnf_0-3

This register repeats 3 times. It parametrized by the rnf_index from 0 to 3. RNF logical list for DDN

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF80 + #{8*[0, 1, 2, 3]}
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

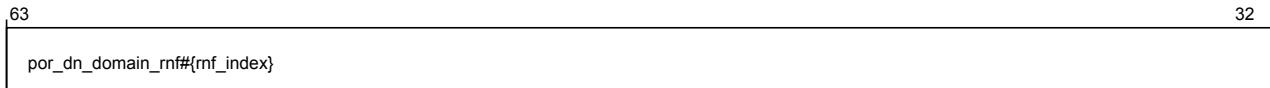


Figure 3-359 por_dn_por_dn_domain_rnf_0-3 (high)

The following table shows the por_dn_domain_rnf_0-3 higher register bit assignments.

Table 3-379 por_dn_por_dn_domain_rnf_0-3 (high)

Bits	Field name	Description	Type	Reset
63:32	por_dn_domain_rnf#{rnf_index}	RNF logical list corresponding to RNF #{{(rnf_index +1)*64)-1};#{rnf_index*64}}	RW	Configuration dependent

The following image shows the lower register bit assignments.

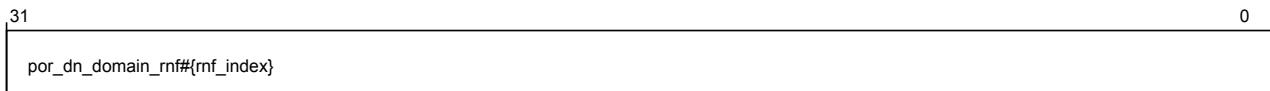


Figure 3-360 por_dn_por_dn_domain_rnf_0-3 (low)

The following table shows the por_dn_domain_rnf_0-3 lower register bit assignments.

Table 3-380 por_dn_por_dn_domain_rnf_0-3 (low)

Bits	Field name	Description	Type	Reset
31:0	por_dn_domain_rnf#{rnf_index}	RNF logical list corresponding to RNF #{{(rnf_index +1)*64)-1};#{rnf_index*64}}	RW	Configuration dependent

por_dn_domain_rnd

RND logical list for DDN

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hFA0
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63

32

por_dn_domain_rnd

Figure 3-361 por_dn_por_dn_domain_rnd (high)

The following table shows the por_dn_domain_rnd higher register bit assignments.

Table 3-381 por_dn_por_dn_domain_rnd (high)

Bits	Field name	Description	Type	Reset
63:32	por_dn_domain_rnd	RND logical list for DDN	RW	Configuration dependent

The following image shows the lower register bit assignments.

31

0

por_dn_domain_rnd

Figure 3-362 por_dn_por_dn_domain_rnd (low)

The following table shows the por_dn_domain_rnd lower register bit assignments.

Table 3-382 por_dn_por_dn_domain_rnd (low)

Bits	Field name	Description	Type	Reset
31:0	por_dn_domain_rnd	RND logical list for DDN	RW	Configuration dependent

por_dn_domain_cxra

CXRA logical list for DDN

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hFA8

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63

32

por_dn_domain_cxra

Figure 3-363 por_dn_por_dn_domain_cxra (high)

The following table shows the por_dn_domain_cxra higher register bit assignments.

Table 3-383 por_dn_por_dn_domain_cxra (high)

Bits	Field name	Description	Type	Reset
63:32	por_dn_domain_cxra	CXRA logical list for DDN	RW	Configuration dependent

The following image shows the lower register bit assignments.

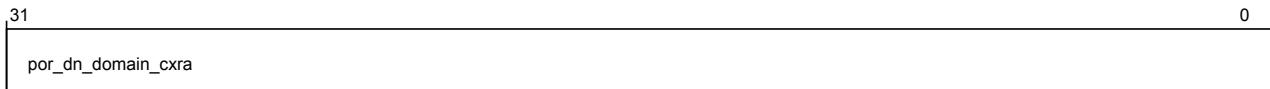


Figure 3-364 por_dn_por_dn_domain_cxra (low)

The following table shows the por_dn_domain_cxra lower register bit assignments.

Table 3-384 por_dn_por_dn_domain_cxra (low)

Bits	Field name	Description	Type	Reset
31:0	por_dn_domain_cxra	CXRA logical list for DDN	RW	Configuration dependent

por_dn_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

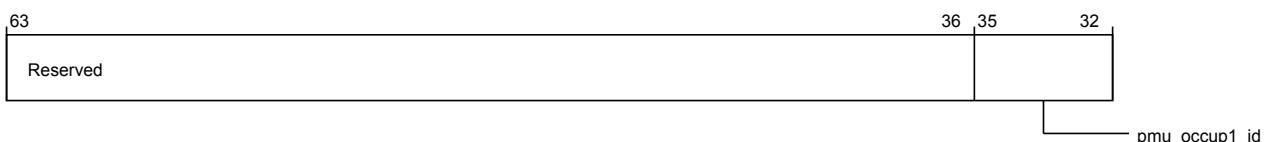


Figure 3-365 por_dn_por_dn_pmu_event_sel (high)

The following table shows the por_dn_pmu_event_sel higher register bit assignments.

Table 3-385 por_dn_por_dn_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:36	Reserved	Reserved	RO	-
35:32	pmu_occup1_id	PMU occupancy event selector ID 4'b0000: All 4'b0001: DVM ops 4'b0010: DVM syncs	RW	4'b0

The following image shows the lower register bit assignments.

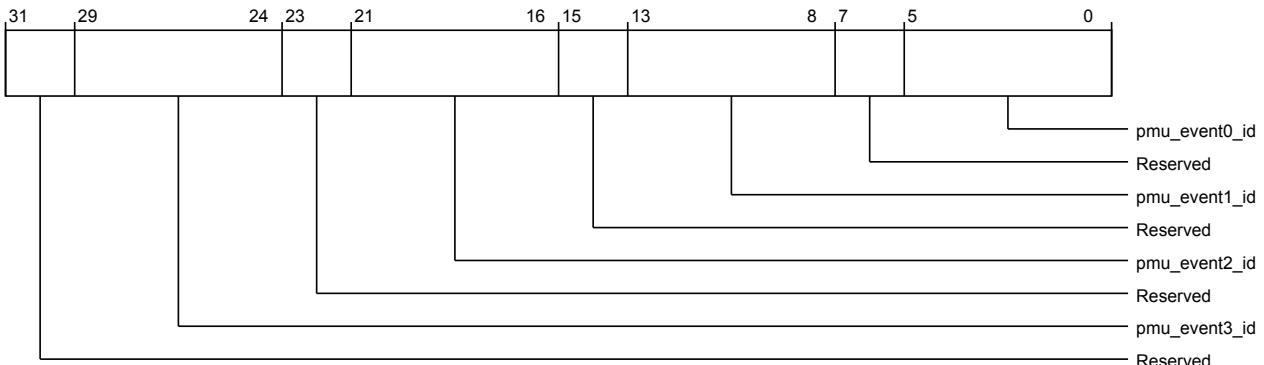


Figure 3-366 por_dn_por_dn_pmu_event_sel (low)

The following table shows the por_dn_pmu_event_sel lower register bit assignments.

Table 3-386 por_dn_por_dn_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	PMU Event 3 ID; see pmu_event0_id for encodings	RW	5'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	PMU Event 2 ID; see pmu_event0_id for encodings	RW	5'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	PMU Event 1 ID; see pmu_event0_id for encodings	RW	5'b0

Table 3-386 por_dn_por_dn_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-
5:0	pmu_event0_id	PMU Event 0 ID 6'h00: No event 6'h01: Number of TLBI DVM op requests 6'h02: Number of BPI DVM op requests 6'h03: Number of PICI DVM op requests 6'h04: Number of VICI DVM op requests 6'h05: Number of DVM sync requests 6'h06: Number of DVM op requests that were filtered using VMID filtering 6'h07: Number of DVM op requests to RNDs, BPI or PICI/VICI, that were filtered 6'h08: Number of retried REQ 6'h09: Number of SNPs sent to RNs 6'h0a: Number of SNPs stalled to RNs due to lack of Crds 6'h0b: DVM tracker full counter 6'h0c: DVM RNF tracker occupancy counter 6'h0d: DVM CXHA tracker occupancy counter 6'h0e: DVM Peer DN tracker occupancy counter 6'h0f: DVM RNF tracker Alloc 6'h10: DVM CXHA tracker Alloc 6'h11: DVM Peer DN tracker Alloc 6'h12: TXSNP stall due to number outstanding limit 6'h13: RXSNP stall starvation threshold hit 6'h14: TXSNP SYNC stall due to outstanding early completed Op	RW	5'b0

3.3.8 CCLA register descriptions

This section lists the DN registers.

por_ccla_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	48 47	32
Reserved	logical_id	

Figure 3-367 por_ccla_por_ccla_node_info (high)

The following table shows the por_ccla_node_info higher register bit assignments.

Table 3-387 por_ccla_por_ccla_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

31	16 15	0
node_id	node_type	

Figure 3-368 por_ccla_por_ccla_node_info (low)

The following table shows the por_ccla_node_info lower register bit assignments.

Table 3-388 por_ccla_por_ccla_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	(CCG_PCIE_ENABLE_PARAM == 1) ? 16'h0106 : 16'h0105

por_ccla_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-369 por_ccla_por_ccla_child_info (high)

The following table shows the por_ccla_child_info higher register bit assignments.

Table 3-389 por_ccla_por_ccla_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

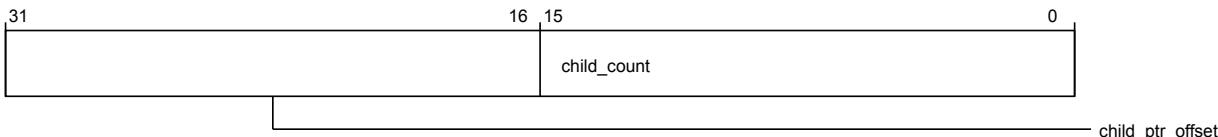


Figure 3-370 por_ccla_por_ccla_child_info (low)

The following table shows the por_ccla_child_info lower register bit assignments.

Table 3-390 por_ccla_por_ccla_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_ccla_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h988

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-371 por_ccla_por_ccla_secure_register_groups_override (high)

The following table shows the por_ccla_secure_register_groups_override higher register bit assignments.

Table 3-391 por_ccla_por_ccla_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

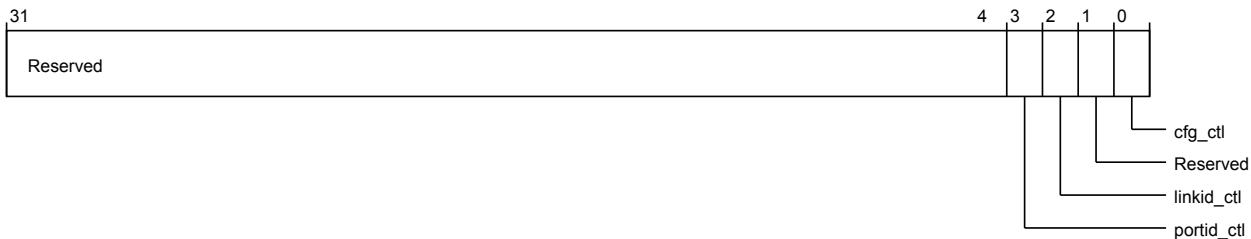


Figure 3-372 por_ccla_por_ccla_secure_register_groups_override (low)

The following table shows the por_ccla_secure_register_groups_override lower register bit assignments.

Table 3-392 por_ccla_por_ccla_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	portid_ctl	Allows non-secure access to secure LA Port ID registers	RW	1'b0
2	linkid_ctl	Allows non-secure access to secure LA Link ID registers	RW	1'b0
1	Reserved	Reserved	RO	-
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_ccla_unit_info

Provides component identification information for CCLA.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h910

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

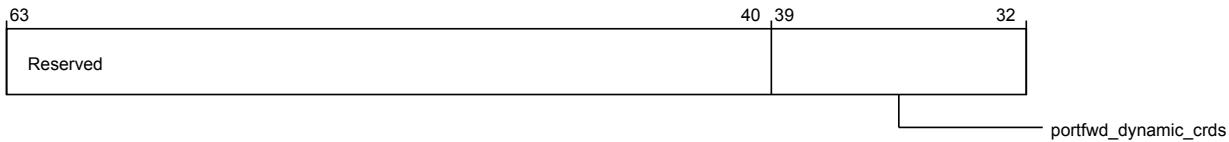


Figure 3-373 por_ccla_por_ccla_unit_info (high)

The following table shows the por_ccla_unit_info higher register bit assignments.

Table 3-393 por_ccla_por_ccla_unit_info (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	portfwd_dynamic_crds	Number of dynamic credits granted by this CCLA port for port forwarded traffic	RO	Configuration dependent

The following image shows the lower register bit assignments.

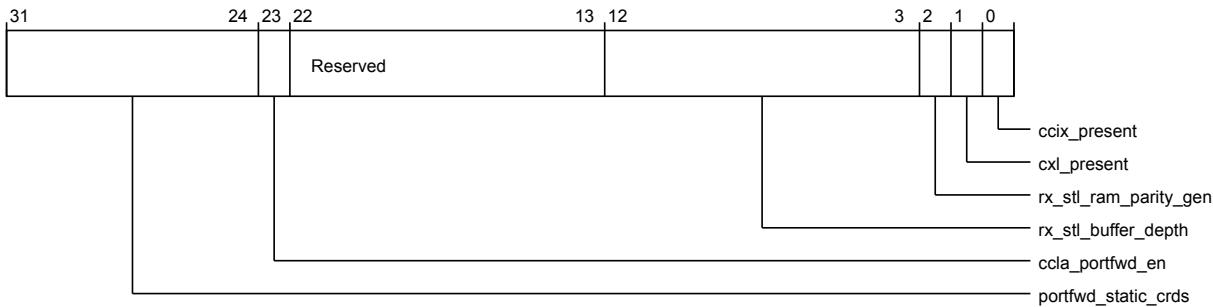


Figure 3-374 por_ccla_por_ccla_unit_info (low)

The following table shows the por_ccla_unit_info lower register bit assignments.

Table 3-394 por_ccla_por_ccla_unit_info (low)

Bits	Field name	Description	Type	Reset
31:24	portfwd_static_crds	Number of static credits granted by this CCLA port for port forwarded traffic	RO	Configuration dependent
23	ccla_portfwd_en	Port forwarding is enabled at this CCLA port	RO	Configuration dependent
22:13	Reserved	Reserved	RO	-
12:3	rx_stl_buffer_depth	Depth of CCL stalling channel RX buffer for CXS RSP with data messages	RO	Configuration dependent
2	rx_stl_ram_parity_gen	Option to generate parity bits for the RX STL buffer	RO	Configuration dependent
1	cxl_present	Option to generate CXL support over CXS	RO	Configuration dependent
0	ccix_present	Option to generate CCIX support over CXS	RO	Configuration dependent

por_ccla_cfg_ctl

Functions as the configuration control register for CCLA.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hB00
Register reset	64'b001001001011100
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccla_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.



Figure 3-375 por_ccla_por_ccla_cfg_ctl (high)

The following table shows the por_ccla_cfg_ctl higher register bit assignments.

Table 3-395 por_ccla_por_ccla_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

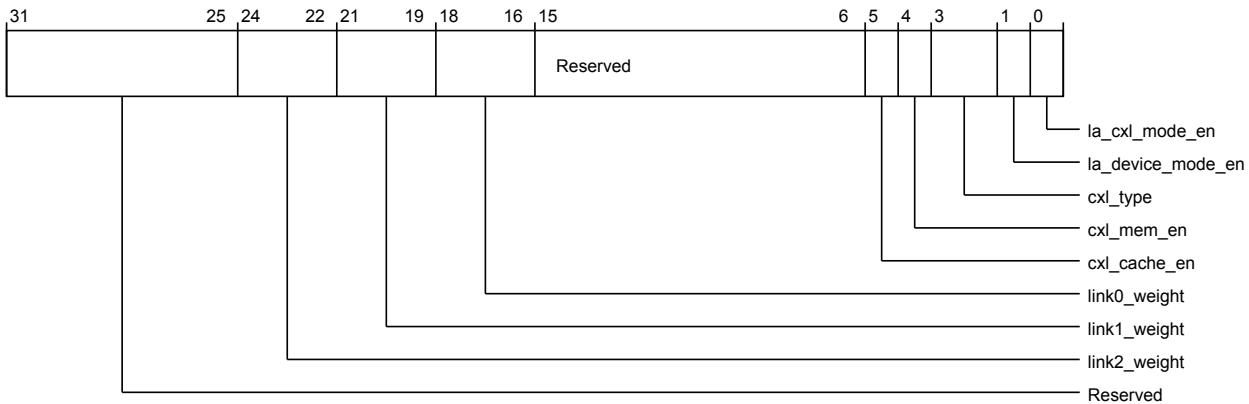


Figure 3-376 por_ccla_por_ccla_cfg_ctl (low)

The following table shows the por_ccla_cfg_ctl lower register bit assignments.

Table 3-396 por_ccla_por_ccla_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24:22	link2_weight	Determines weight of link2 to start forming a TLP in presence of pending messages to other links; applies for message packing	RW	3'b001
21:19	link1_weight	Determines weight of link1 to start forming a TLP in presence of pending messages to other links; applies for message packing	RW	3'b001
18:16	link0_weight	Determines weight of link0 to start forming a TLP in presence of pending messages to other links; applies for message packing	RW	3'b001
15:6	Reserved	Reserved	RO	-
5	cxl_cache_en	Enable CXL .cache mode, by default is disabled	RW	1'b0
4	cxl_mem_en	Enable CXL .mem mode, by default is enabled	RW	1'b1
3:2	cxl_type	Used to program CXL Type 2'b00: Reserved 2'b01: Type1 2'b10: Type2 2'b11: Type3	RW	2'b11
1	la_device_mode_en	Enable the Device mode, by default set to Host mode	RW	1'b0
0	la_cxl_mode_en	When set enables CXL mode	RW	1'b0

por_ccla_aux_ctl

Functions as the auxiliary control register for CCLA.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hB08
Register reset	64'b01001001001001001001001001010
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

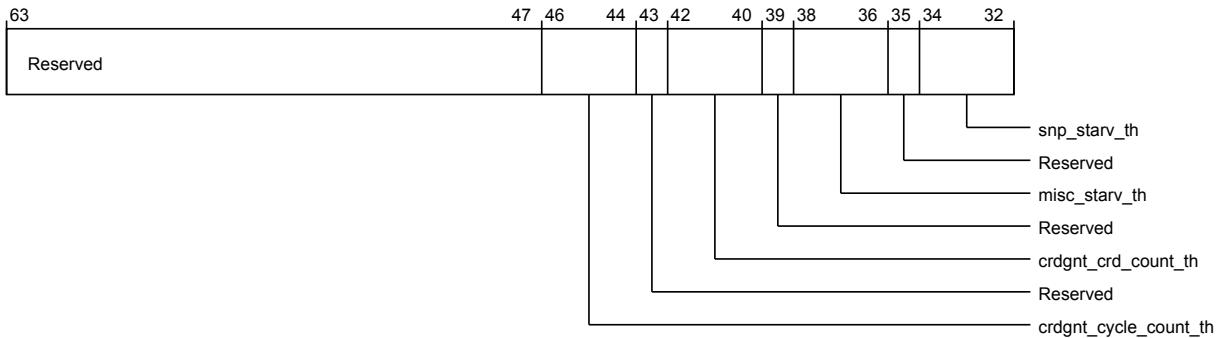


Figure 3-377 por_ccla_por_ccla_aux_ctl (high)

The following table shows the por_ccla_aux_ctl higher register bit assignments.

Table 3-397 por_ccla_por_ccla_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:44	crdgnt_cycle_count_th	Maximum number of cycles that need to be elapsed since the end of previous TLP to send a credit grant message 3'b000: 32 cycles 3'b001: 64 cycles 3'b010: 128 cycles 3'b011: 256 cycles	RW	3'b010
43	Reserved	Reserved	RO	-
42:40	crdgnt_crd_count_th	Maximum number of credits that need to be accumulated to send a credit grant message 3'b000: 16 credits 3'b001: 32 credits 3'b010: 64 credits 3'b011: 128 credits	RW	3'b010
39	Reserved	Reserved	RO	-
38:36	misc_starv_th	Maximum number of consecutive instances a Misc message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010

Table 3-397 por_ccla_por_ccla_aux_ctl (high) (continued)

Bits	Field name	Description	Type	Reset
35	Reserved	Reserved	RO	-
34:32	snp_starv_th	Maximum number of consecutive instances a Snoop Request message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010

The following image shows the lower register bit assignments.

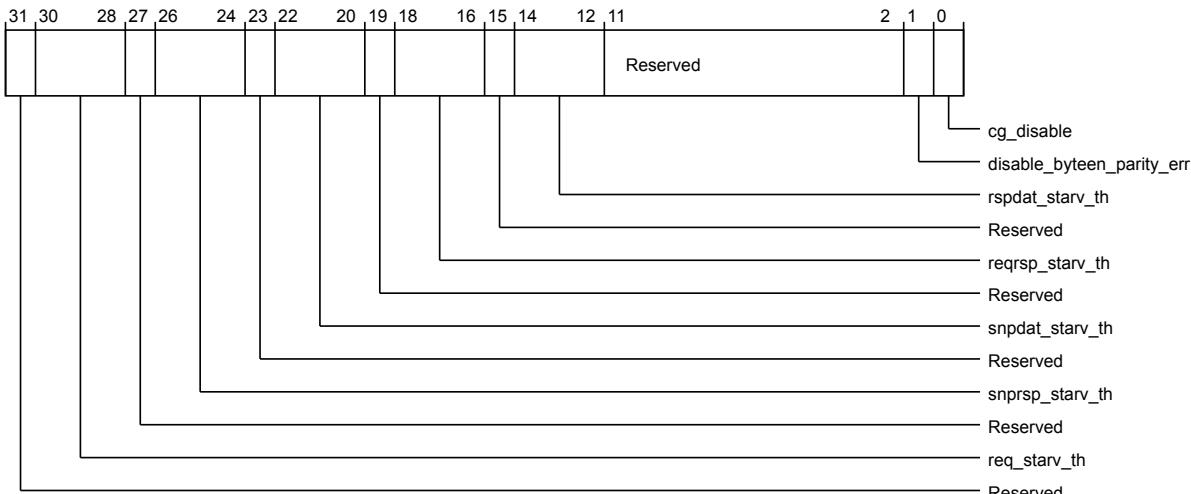


Figure 3-378 por_ccla_por_ccla_aux_ctl (low)

The following table shows the por_ccla_aux_ctl lower register bit assignments.

Table 3-398 por_ccla_por_ccla_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	req_starv_th	Maximum number of consecutive instances a Memory Request message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010

Table 3-398 por_ccla_por_ccla_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
27	Reserved	Reserved	RO	-
26:24	snprsp_starv_th	Maximum number of consecutive instances a Snoop Response without Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
23	Reserved	Reserved	RO	-
22:20	snpdat_starv_th	Maximum number of consecutive instances a Snoop Response with Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
19	Reserved	Reserved	RO	-
18:16	reqrsp_starv_th	Maximum number of consecutive instances a Memory Response without Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
15	Reserved	Reserved	RO	-
14:12	rspdat_starv_th	Maximum number of consecutive instances a Memory Response with Data message loses to other message types in forming a TLP 3'b000: 8 cycles 3'b001: 16 cycles 3'b010: 32 cycles 3'b011: 64 cycles 3'b100: 128 cycles	RW	3'b010
11:2	Reserved	Reserved	RO	-

Table 3-398 por_ccla_por_ccla_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
1	disable_byteen_parity_err	Disables CCLA RX RAM byte enable parity errors	RW	1'b1
0	cg_disable	Disables CCLA architectural clock gates	RW	1'b0

por_ccla_ccix_prop_capabilities

Contains CCIX-supported properties.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hC00

Register reset 64'b001001

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-379 por_ccla_por_ccla_ccix_prop_capabilities (high)

The following table shows the por_ccla_ccix_prop_capabilities higher register bit assignments.

Table 3-399 por_ccla_por_ccla_ccix_prop_capabilities (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

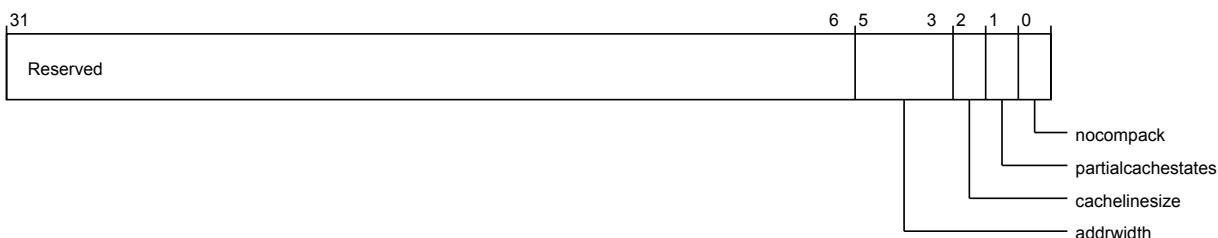


Figure 3-380 por_ccla_por_ccla_ccix_prop_capabilities (low)

The following table shows the por_ccla_ccix_prop_capabilities lower register bit assignments.

Table 3-400 por_ccla_por_ccla_ccix_prop_capabilities (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5:3	addrwidth	Address width supported 3'b000: 48b 3'b001: 52b 3'b010: 56b 3'b011: 60b 3'b100: 64b	RO	3'b001
2	cachelinesize	Cacheline size supported 1'b0: 64B 1'b1: 128B	RO	1'b0
1	partialcachestates	Partial cache states supported 1'b0: False 1'b1: True	RO	1'b0
0	nocompack	No CompAck supported 1'b0: False 1'b1: True	RO	1'b1

por_ccla_cxs_attr_capabilities

Contains CXS supported attributes.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hC08

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

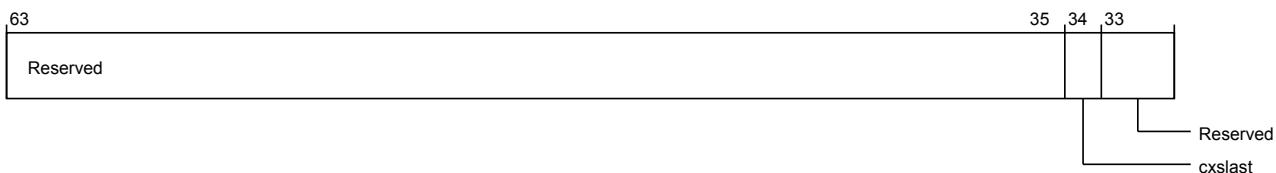


Figure 3-381 por_ccla_por_ccla_cxs_attr_capabilities (high)

The following table shows the por_ccla_cxs_attr_capabilities higher register bit assignments.

Table 3-401 por_ccla_por_ccla_cxs_attr_capabilities (high)

Bits	Field name	Description	Type	Reset
63:35	Reserved	Reserved	RO	-
34	cxslast	CXS LAST signal is supported	RO	Configuration dependent
33:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

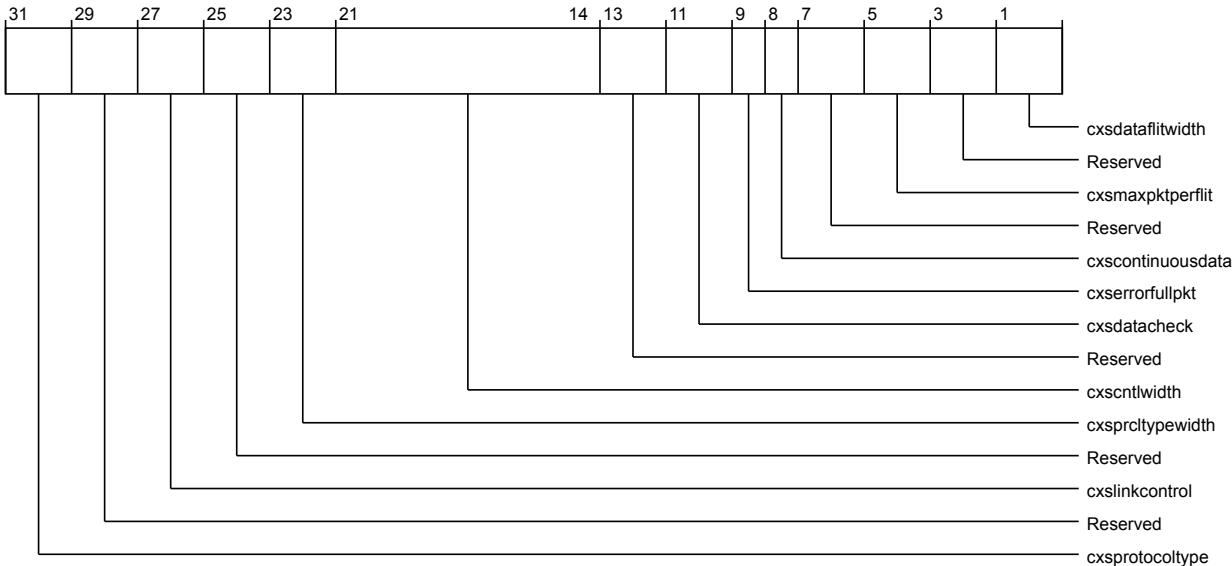


Figure 3-382 por_ccla_por_ccla_cxs_attr_capabilities (low)

The following table shows the por_ccla_cxs_attr_capabilities lower register bit assignments.

Table 3-402 por_ccla_por_ccla_cxs_attr_capabilities (low)

Bits	Field name	Description	Type	Reset
31:30	cxspctrltype	CXS Protocol type signal is supported	RO	Configuration dependent
29:28	Reserved	Reserved	RO	-
27:26	cxslinkcontrol	Set to Explicit Credit Return.	RO	Configuration dependent
25:24	Reserved	Reserved	RO	-
23:22	cxspctrltypewidth	Width of CXS TX/RX control	RO	Configuration dependent
21:14	cxscntlwidth	Width of CXS TX/RX control	RO	Configuration dependent
13:12	Reserved	Reserved	RO	-

Table 3-402 por_ccla_por_ccla_cxs_attr_capabilities (low) (continued)

Bits	Field name	Description	Type	Reset
11:10	cxsdatacheck	CXS datacheck supported 2'b00: None 2'b01: Parity 2'b10: SECDED	RO	Configuration dependent
9	cxserrorfullpkt	CXS error full packet supported 1'b0: False 1'b1: True	RO	Configuration dependent
8	cxscontinuousdata	CXS continuous data supported 1'b0: False 1'b1: True	RO	Configuration dependent
7:6	Reserved	Reserved	RO	-
5:4	cxsmaxpktperflit	CXS maximum packets per flit supported 2'b00: 2 2'b01: 3 2'b10: 4	RO	Configuration dependent
3:2	Reserved	Reserved	RO	-
1:0	cxsdataflitwidth	CXS data flit width supported 2'b00: 256b 2'b01: 512b 2'b10: 1024b	RO	2'b01

por_ccla_permmsg_pyId_0_63

Contains bits[63:0] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD00

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

per_msg_pyld_0_63

Figure 3-383 por_ccla_por_ccla_permsg_pyld_0_63 (high)

The following table shows the por_ccla_permsg_pyld_0_63 higher register bit assignments.

Table 3-403 por_ccla_por_ccla_permsg_pyld_0_63 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_0_63	Protocol Error Msg Payload[63:0]	RW	64'b0

The following image shows the lower register bit assignments.

31

0

per_msg_pyld_0_63

Figure 3-384 por_ccla_por_ccla_permsg_pyld_0_63 (low)

The following table shows the por_ccla_permsg_pyld_0_63 lower register bit assignments.

Table 3-404 por_ccla_por_ccla_permsg_pyld_0_63 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_0_63	Protocol Error Msg Payload[63:0]	RW	64'b0

por_ccla_permsg_pyld_64_127

Contains bits[127:64] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD08

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

per_msg_pyld_64_127

Figure 3-385 por_ccla_por_ccla_permsg_pyld_64_127 (high)

The following table shows the por_ccla_permsg_pyld_64_127 higher register bit assignments.

Table 3-405 por_ccla_por_ccla_permmsg_pyld_64_127 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_64_127	Protocol Error Msg Payload[127:64]	RW	64'b0

The following image shows the lower register bit assignments.

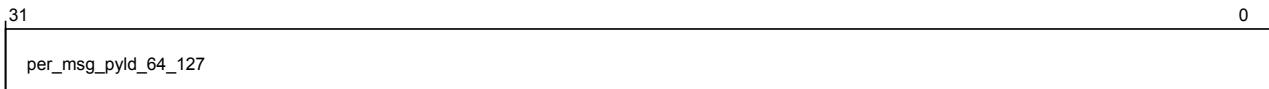


Figure 3-386 por_ccla_por_ccla_permmsg_pyld_64_127 (low)

The following table shows the por_ccla_permmsg_pyld_64_127 lower register bit assignments.

Table 3-406 por_ccla_por_ccla_permmsg_pyld_64_127 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_64_127	Protocol Error Msg Payload[127:64]	RW	64'b0

por_ccla_permmsg_pyld_128_191

Contains bits[192:128] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD10

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-387 por_ccla_por_ccla_permmsg_pyld_128_191 (high)

The following table shows the por_ccla_permmsg_pyld_128_191 higher register bit assignments.

Table 3-407 por_ccla_por_ccla_permmsg_pyld_128_191 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_128_191	Protocol Error Msg Payload[191:128]	RW	64'b0

The following image shows the lower register bit assignments.

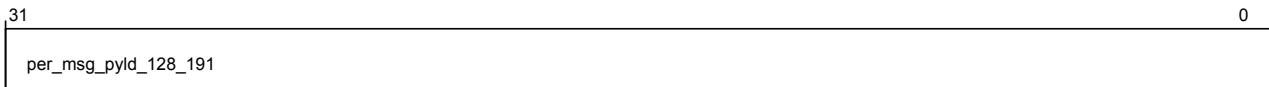


Figure 3-388 por_ccla_por_ccla_permsg_pyld_128_191 (low)

The following table shows the por_ccla_permsg_pyld_128_191 lower register bit assignments.

Table 3-408 por_ccla_por_ccla_permsg_pyld_128_191 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_128_191	Protocol Error Msg Payload[191:128]	RW	64'b0

por_ccla_permsg_pyld_192_255

Contains bits[255:192] of CCIX Protocol Error (PER) Message payload.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD18

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

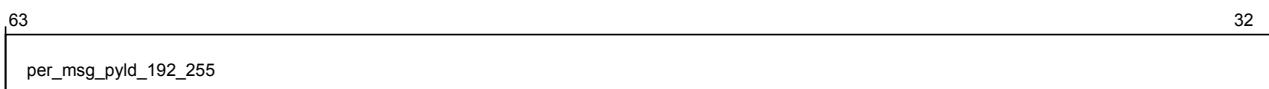


Figure 3-389 por_ccla_por_ccla_permsg_pyld_192_255 (high)

The following table shows the por_ccla_permsg_pyld_192_255 higher register bit assignments.

Table 3-409 por_ccla_por_ccla_permsg_pyld_192_255 (high)

Bits	Field name	Description	Type	Reset
63:32	per_msg_pyld_192_255	Protocol Error Msg Payload[255:192]	RW	64'b0

The following image shows the lower register bit assignments.

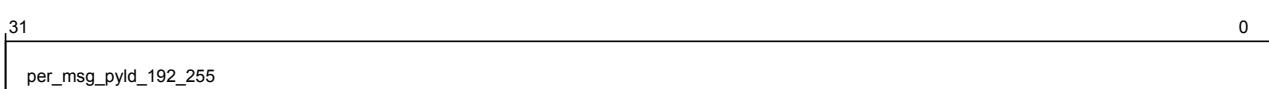


Figure 3-390 por_ccla_por_ccla_permsg_pyld_192_255 (low)

The following table shows the por_ccla_permsg_pyld_192_255 lower register bit assignments.

Table 3-410 por_ccla_por_ccla_permmsg_pyld_192_255 (low)

Bits	Field name	Description	Type	Reset
31:0	per_msg_pyld_192_255	Protocol Error Msg Payload[255:192]	RW	64'b0

por_ccla_permmsg_ctl

Contains Control bits to trigger CCIX Protocol Error (PER) Message.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD20

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

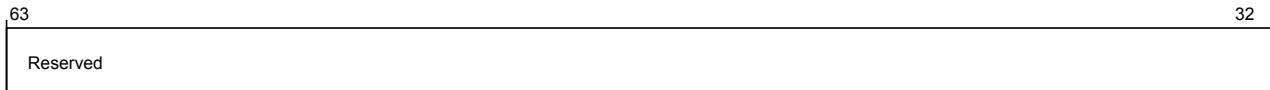


Figure 3-391 por_ccla_por_ccla_permmsg_ctl (high)

The following table shows the por_ccla_permmsg_ctl higher register bit assignments.

Table 3-411 por_ccla_por_ccla_permmsg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

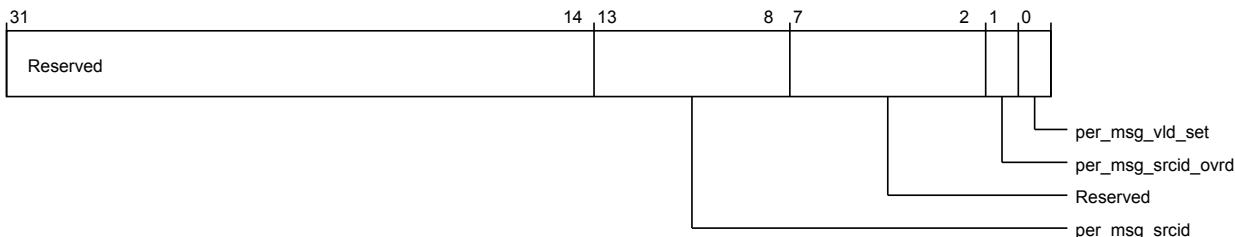


Figure 3-392 por_ccla_por_ccla_permmsg_ctl (low)

The following table shows the por_ccla_permmsg_ctl lower register bit assignments.

Table 3-412 por_ccla_por_ccla_permmsg_ctl (low)

Bits	Field name	Description	Type	Reset
31:14	Reserved	Reserved	RO	-
13:8	per_msg_srcid	Contains Source ID used on CCIX Protocol Error Msg. Used when per_msg_srcid_ovrd is set.	RW	6'b0
7:2	Reserved	Reserved	RO	-
1	per_msg_srcid_ovrd	When set, overrides the Source ID on Protocol Error Msg by value specified in this register. Or else the source ID from payload[55:48] is used.	RW	1'b0
0	per_msg_vld_set	When set, sends CCIX Protocol Error Msg. Must be cleared after the current error is processed and before a new error message is triggered	RW	1'b0

por_ccla_err_agent_id

Contains Error Agent ID. Must be programmed by CCIX discovery s/w. Used as TargetID on CCIX Protocol Error (PER) Message.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD28

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-393 por_ccla_por_ccla_err_agent_id (high)

The following table shows the por_ccla_err_agent_id higher register bit assignments.

Table 3-413 por_ccla_por_ccla_err_agent_id (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

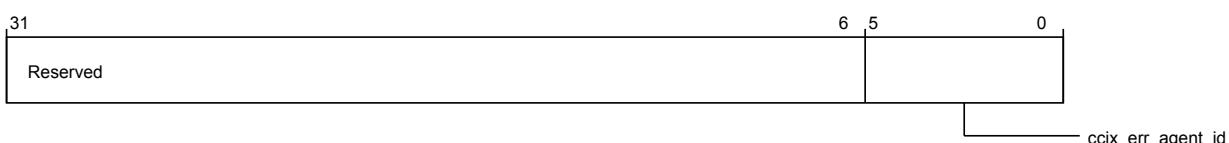


Figure 3-394 por_ccla_por_ccla_err_agent_id (low)

The following table shows the por_ccla_err_agent_id lower register bit assignments.

Table 3-414 por_ccla_por_ccla_err_agent_id (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5:0	ccix_err_agent_id	CCIX Error AgentID	RW	6'b0

por_ccla_agentid_to_portid_reg0

Specifies the mapping of Agent ID to Port ID for Agent IDs 0 to 7.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_ccla_secure_register_groups_override.portid_ctl

Table 3.11 A comparison of the 1990-1991 and 1991-1992 seasons.

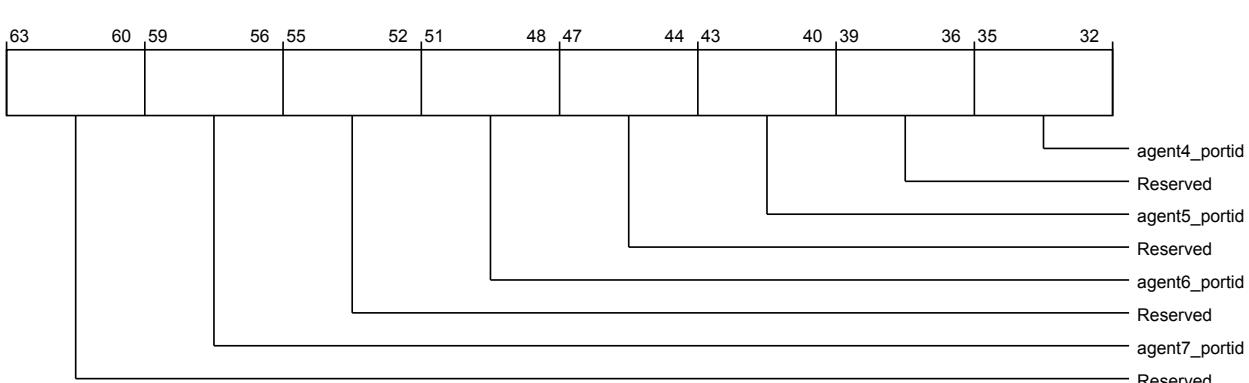


Figure 3-395 por ccla por ccla agentid to portid req0 (high)

The following table shows the port class assignment to portid register bit assignments.

Table 3-415 por ccla por ccla agentid to portid req0 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent7_portid	Specifies the Port ID for Agent ID 7	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent6_portid	Specifies the Port ID for Agent ID 6	RW	4'h0
47:44	Reserved	Reserved	RO	-

Table 3-415 por_ccla_por_ccla_agentid_to_portid_reg0 (high) (continued)

Bits	Field name	Description	Type	Reset
43:40	agent5_portid	Specifies the Port ID for Agent ID 5	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent4_portid	Specifies the Port ID for Agent ID 4	RW	4'h0

The following image shows the lower register bit assignments.

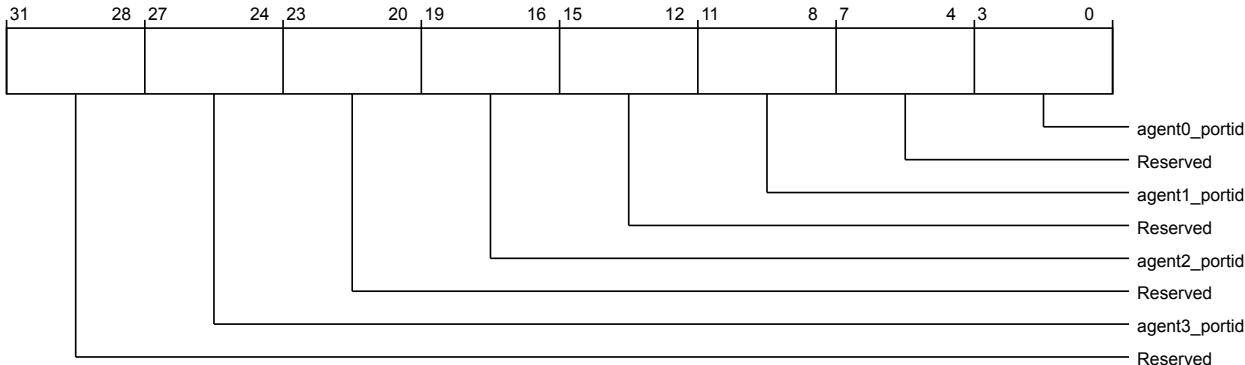


Figure 3-396 por_ccla_por_ccla_agentid_to_portid_reg0 (low)

The following table shows the por_ccla_agentid_to_portid_reg0 lower register bit assignments.

Table 3-416 por_ccla_por_ccla_agentid_to_portid_reg0 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent3_portid	Specifies the Port ID for Agent ID 3	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent2_portid	Specifies the Port ID for Agent ID 2	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent1_portid	Specifies the Port ID for Agent ID 1	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent0_portid	Specifies the Port ID for Agent ID 0	RW	4'h0

por_ccla_agentid_to_portid_reg1

Specifies the mapping of Agent ID to Port ID for Agent IDs 8 to 15.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD38

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

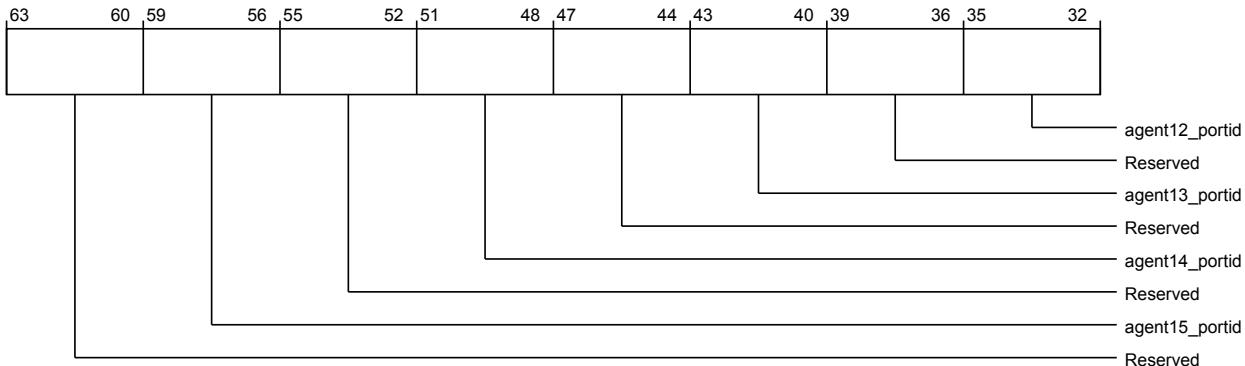


Figure 3-397 por_ccla_por_ccla_agentid_to_portid_reg1 (high)

The following table shows the por_ccla_agentid_to_portid_reg1 higher register bit assignments.

Table 3-417 por_ccla_por_ccla_agentid_to_portid_reg1 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent15_portid	Specifies the Port ID for Agent ID 15	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent14_portid	Specifies the Port ID for Agent ID 14	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent13_portid	Specifies the Port ID for Agent ID 13	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent12_portid	Specifies the Port ID for Agent ID 12	RW	4'h0

The following image shows the lower register bit assignments.

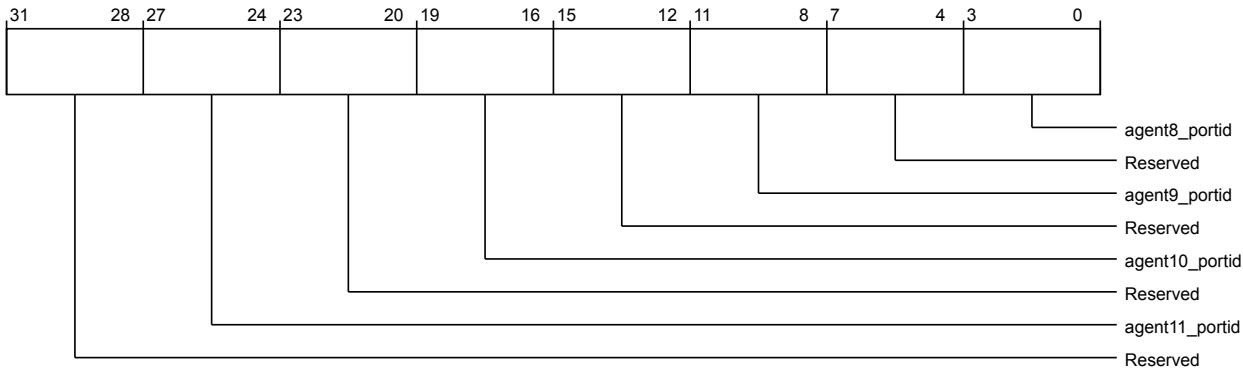


Figure 3-398 por_ccla_por_ccla_agentid_to_portid_reg1 (low)

The following table shows the por_ccla_agentid_to_portid_reg1 lower register bit assignments.

Table 3-418 por_ccla_por_ccla_agentid_to_portid_reg1 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent11_portid	Specifies the Port ID for Agent ID 11	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent10_portid	Specifies the Port ID for Agent ID 10	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent9_portid	Specifies the Port ID for Agent ID 9	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent8_portid	Specifies the Port ID for Agent ID 8	RW	4'h0

por_ccla_agentid_to_portid_reg2

Specifies the mapping of Agent ID to Port ID for Agent IDs 16 to 23.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

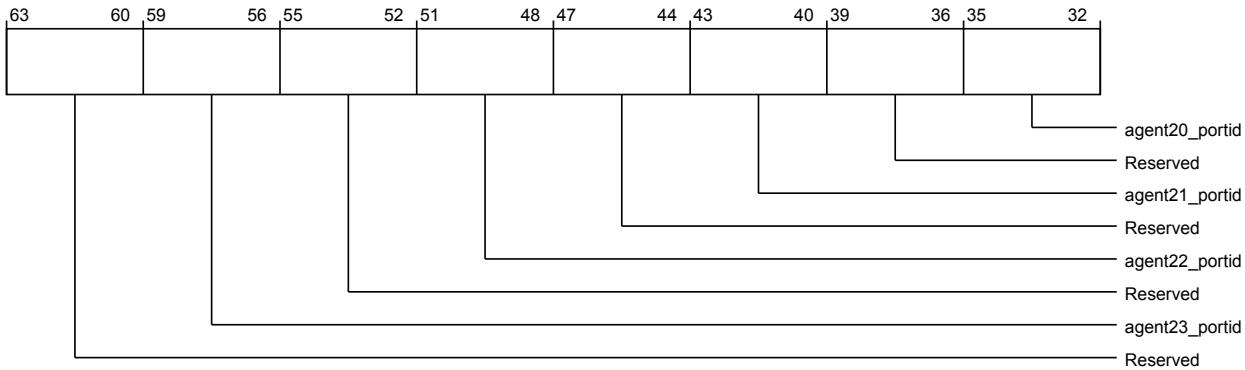


Figure 3-399 por_ccla_por_ccla_agentid_to_portid_reg2 (high)

The following table shows the por_ccla_agentid_to_portid_reg2 higher register bit assignments.

Table 3-419 por_ccla_por_ccla_agentid_to_portid_reg2 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent23_portid	Specifies the Port ID for Agent ID 23	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent22_portid	Specifies the Port ID for Agent ID 22	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent21_portid	Specifies the Port ID for Agent ID 21	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent20_portid	Specifies the Port ID for Agent ID 20	RW	4'h0

The following image shows the lower register bit assignments.

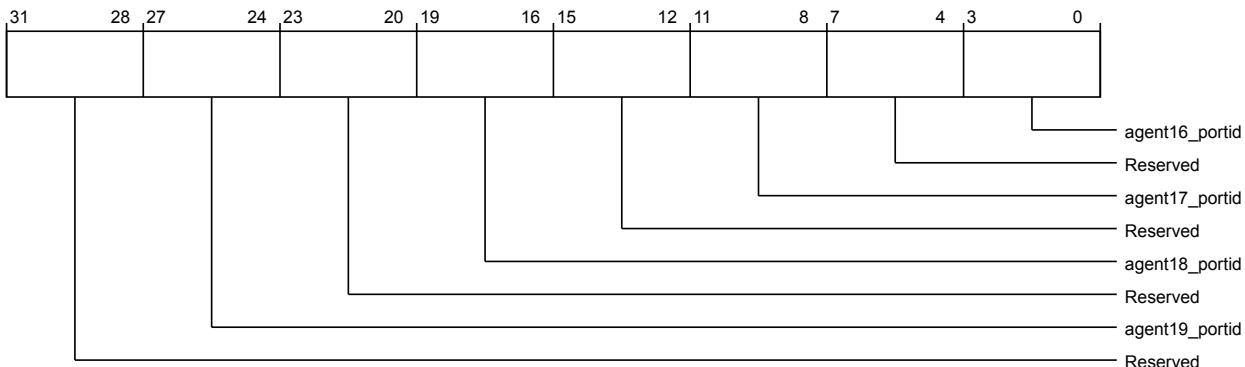


Figure 3-400 por_ccla_por_ccla_agentid_to_portid_reg2 (low)

The following table shows the por_ccla_agentid_to_portid_reg2 lower register bit assignments.

Table 3-420 por_ccla_por_ccla_agentid_to_portid_reg2 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent19_portid	Specifies the Port ID for Agent ID 19	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent18_portid	Specifies the Port ID for Agent ID 18	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent17_portid	Specifies the Port ID for Agent ID 17	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent16_portid	Specifies the Port ID for Agent ID 16	RW	4'h0

por_ccla_agentid_to_portid_reg3

Specifies the mapping of Agent ID to Port ID for Agent IDs 24 to 31.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

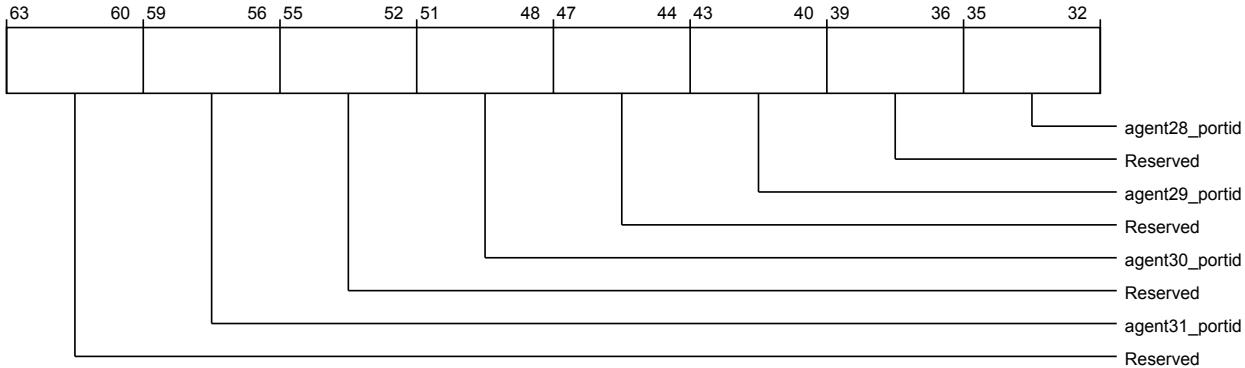


Figure 3-401 por_ccla_por_ccla_agentid_to_portid_reg3 (high)

The following table shows the por_ccla_agentid_to_portid_reg3 higher register bit assignments.

Table 3-421 por_ccla_por_ccla_agentid_to_portid_reg3 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent31_portid	Specifies the Port ID for Agent ID 31	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent30_portid	Specifies the Port ID for Agent ID 30	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent29_portid	Specifies the Port ID for Agent ID 29	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent28_portid	Specifies the Port ID for Agent ID 28	RW	4'h0

The following image shows the lower register bit assignments.

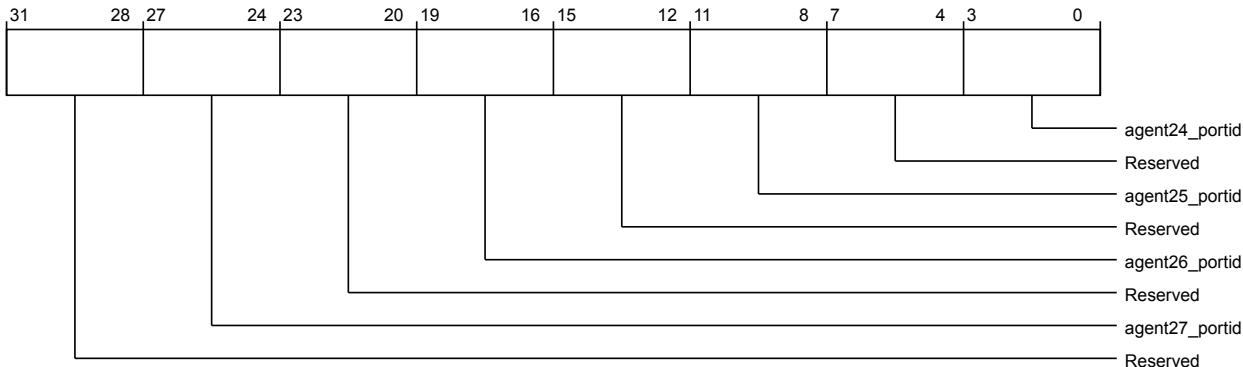


Figure 3-402 por_ccla_por_ccla_agentid_to_portid_reg3 (low)

The following table shows the por_ccla_agentid_to_portid_reg3 lower register bit assignments.

Table 3-422 por_ccla_por_ccla_agentid_to_portid_reg3 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent27_portid	Specifies the Port ID for Agent ID 27	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent26_portid	Specifies the Port ID for Agent ID 26	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent25_portid	Specifies the Port ID for Agent ID 25	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent24_portid	Specifies the Port ID for Agent ID 24	RW	4'h0

por_ccla_agentid_to_portid_reg4

Specifies the mapping of Agent ID to Port ID for Agent IDs 32 to 39.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD50
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

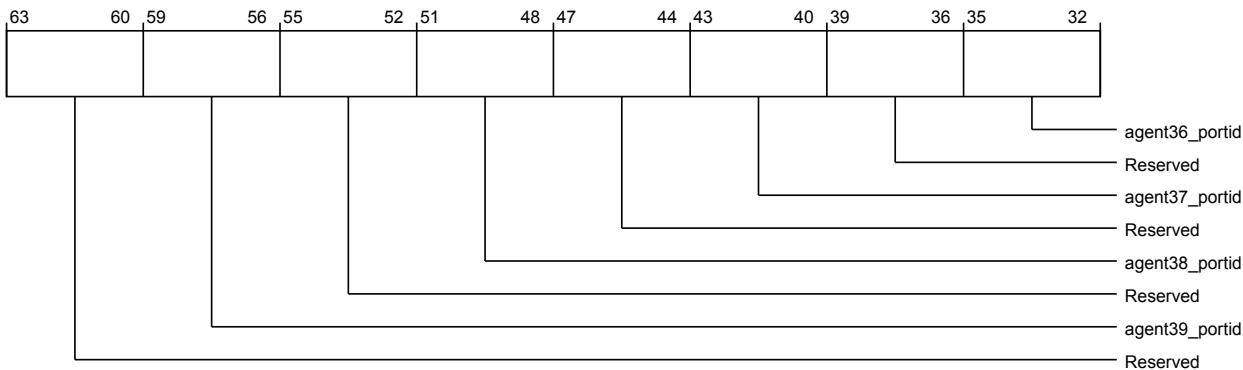


Figure 3-403 por_ccla_por_ccla_agentid_to_portid_reg4 (high)

The following table shows the por_ccla_agentid_to_portid_reg4 higher register bit assignments.

Table 3-423 por_ccla_por_ccla_agentid_to_portid_reg4 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent39_portid	Specifies the Port ID for Agent ID 39	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent38_portid	Specifies the Port ID for Agent ID 38	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent37_portid	Specifies the Port ID for Agent ID 37	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent36_portid	Specifies the Port ID for Agent ID 36	RW	4'h0

The following image shows the lower register bit assignments.

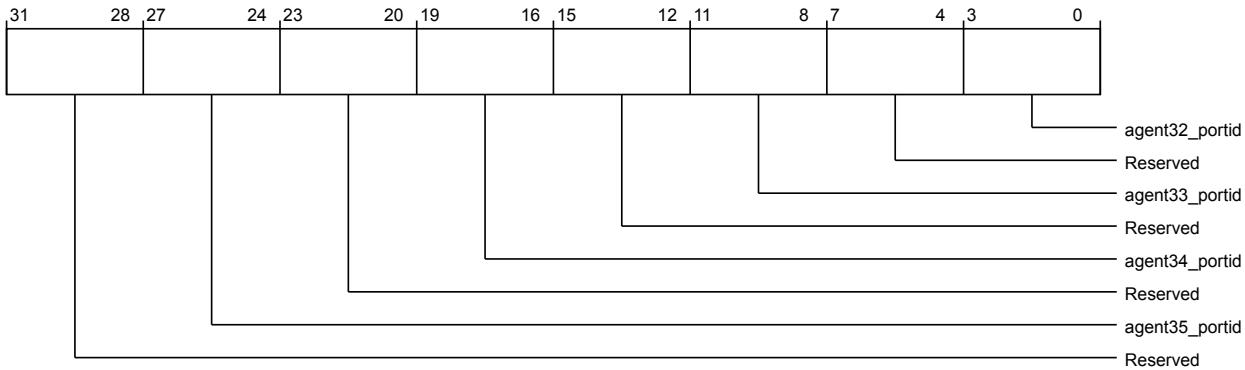


Figure 3-404 por_ccla_por_ccla_agentid_to_portid_reg4 (low)

The following table shows the por_ccla_agentid_to_portid_reg4 lower register bit assignments.

Table 3-424 por_ccla_por_ccla_agentid_to_portid_reg4 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent35_portid	Specifies the Port ID for Agent ID 35	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent34_portid	Specifies the Port ID for Agent ID 34	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent33_portid	Specifies the Port ID for Agent ID 33	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent32_portid	Specifies the Port ID for Agent ID 32	RW	4'h0

por_ccla_agentid_to_portid_reg5

Specifies the mapping of Agent ID to Port ID for Agent IDs 40 to 47.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD58
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

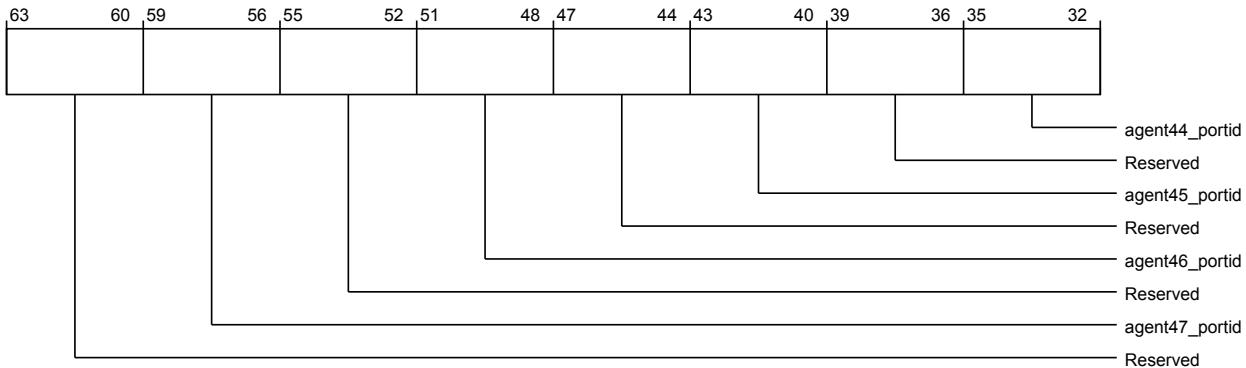


Figure 3-405 por_ccla_por_ccla_agentid_to_portid_reg5 (high)

The following table shows the portid register bit assignments.

Table 3-425 por_ccla_por_ccla_agentid_to_portid_reg5 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent47_portid	Specifies the Port ID for Agent ID 47	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent46_portid	Specifies the Port ID for Agent ID 46	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent45_portid	Specifies the Port ID for Agent ID 45	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent44_portid	Specifies the Port ID for Agent ID 44	RW	4'h0

The following image shows the lower register bit assignments.

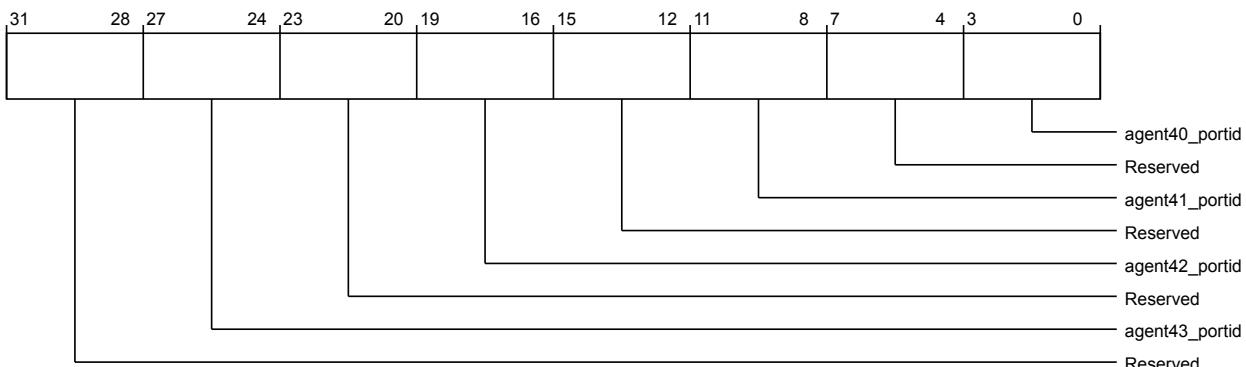


Figure 3-406 por_ccla_por ccla agentid to portid reg5 (low)

The following table shows the portid to reg5 lower register bit assignments.

Table 3-426 por_ccla_por_ccla_agentid_to_portid_reg5 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent43_portid	Specifies the Port ID for Agent ID 43	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent42_portid	Specifies the Port ID for Agent ID 42	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent41_portid	Specifies the Port ID for Agent ID 41	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent40_portid	Specifies the Port ID for Agent ID 40	RW	4'h0

por_ccla_agentid_to_portid_reg6

Specifies the mapping of Agent ID to Port ID for Agent IDs 48 to 55.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD60

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_ccla_secure_register_groups_override.portid_ctl

The following table summarizes the main findings.

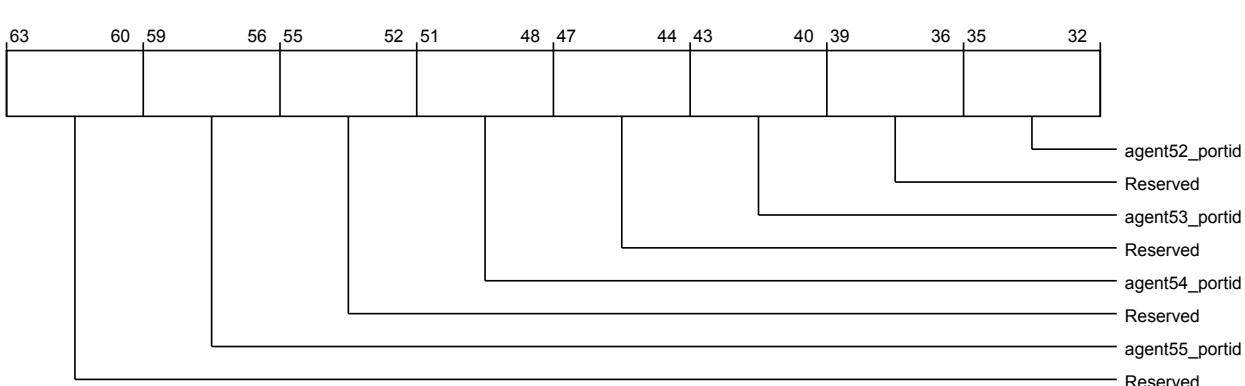


Figure 3-407 por_ccla_por_ccla_agentid_to_portid_reg6 (high)

The following table shows the port class agentid to portid reg6 higher register bit assignments.

Table 3-427 por_ccla_por_ccla_agentid_to_portid_reg6 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent55_portid	Specifies the Port ID for Agent ID 55	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent54_portid	Specifies the Port ID for Agent ID 54	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent53_portid	Specifies the Port ID for Agent ID 53	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent52_portid	Specifies the Port ID for Agent ID 52	RW	4'h0

The following image shows the lower register bit assignments.

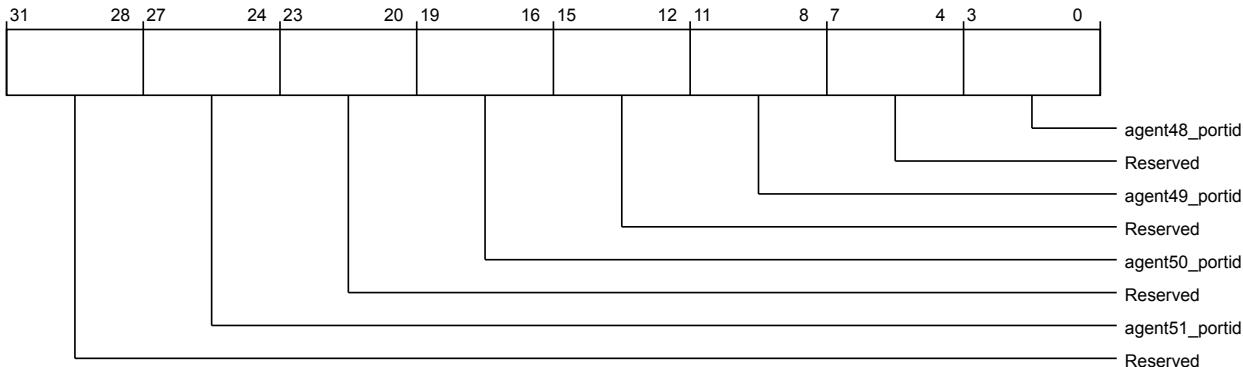


Figure 3-408 por_ccla_por_ccla_agentid_to_portid_reg6 (low)

The following table shows the por_ccla_agentid_to_portid_reg6 lower register bit assignments.

Table 3-428 por_ccla_por_ccla_agentid_to_portid_reg6 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent51_portid	Specifies the Port ID for Agent ID 51	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent50_portid	Specifies the Port ID for Agent ID 50	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent49_portid	Specifies the Port ID for Agent ID 49	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent48_portid	Specifies the Port ID for Agent ID 48	RW	4'h0

por_ccla_agentid_to_portid_reg7

Specifies the mapping of Agent ID to Port ID for Agent IDs 56 to 63.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD68
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

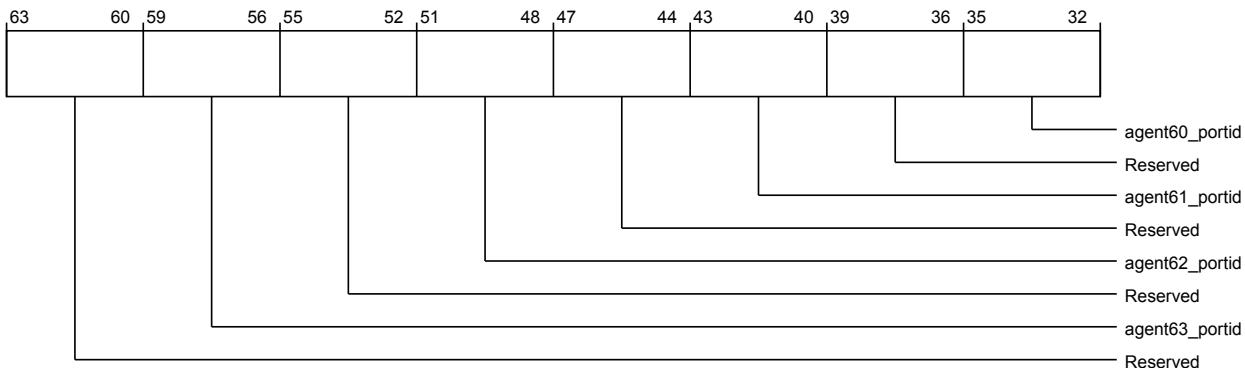


Figure 3-409 por_ccla_por_ccla_agentid_to_portid_reg7 (high)

The following table shows the portid register bit assignments.

Table 3-429 por_ccla_por_ccla_agentid_to_portid_reg7 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:56	agent63_portid	Specifies the Port ID for Agent ID 63	RW	4'h0
55:52	Reserved	Reserved	RO	-
51:48	agent62_portid	Specifies the Port ID for Agent ID 62	RW	4'h0
47:44	Reserved	Reserved	RO	-
43:40	agent61_portid	Specifies the Port ID for Agent ID 61	RW	4'h0
39:36	Reserved	Reserved	RO	-
35:32	agent60_portid	Specifies the Port ID for Agent ID 60	RW	4'h0

The following image shows the lower register bit assignments.

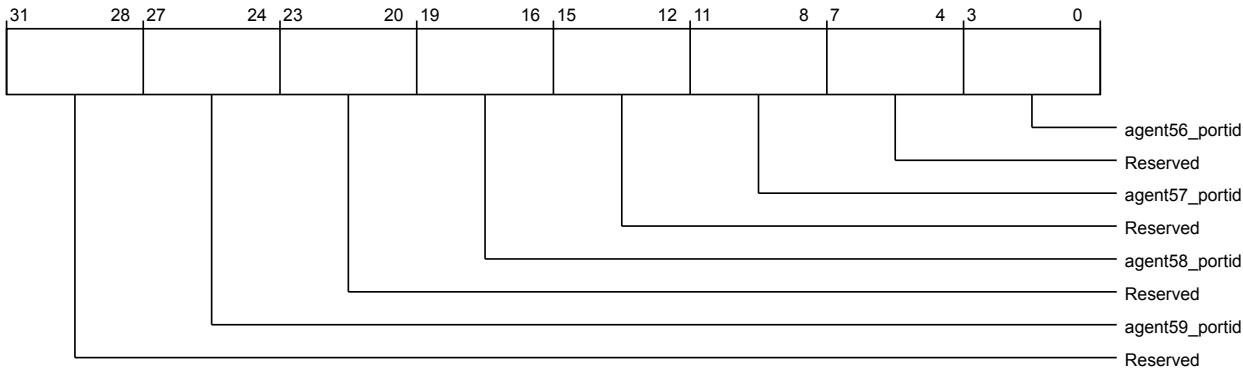


Figure 3-410 por_ccla_por_ccla_agentid_to_portid_reg7 (low)

The following table shows the por_ccla_agentid_to_portid_reg7 lower register bit assignments.

Table 3-430 por_ccla_por_ccla_agentid_to_portid_reg7 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	agent59_portid	Specifies the Port ID for Agent ID 59	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	agent58_portid	Specifies the Port ID for Agent ID 58	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	agent57_portid	Specifies the Port ID for Agent ID 57	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	agent56_portid	Specifies the Port ID for Agent ID 56	RW	4'h0

por_ccla_agentid_to_portid_val

Specifies which Agent ID to Port ID mappings are valid.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD70

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccla_secure_register_groups_override.portid_ctl

The following image shows the higher register bit assignments.

63	valid	32
----	-------	----

Figure 3-411 por_ccla_por_ccla_agentid_to_portid_val (high)

The following table shows the por_ccla_agentid_to_portid_val higher register bit assignments.

Table 3-431 por_ccla_por_ccla_agentid_to_portid_val (high)

Bits	Field name	Description	Type	Reset
63:32	valid	Specifies whether the Port ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

The following image shows the lower register bit assignments.

31	valid	0
----	-------	---

Figure 3-412 por_ccla_por_ccla_agentid_to_portid_val (low)

The following table shows the por_ccla_agentid_to_portid_val lower register bit assignments.

Table 3-432 por_ccla_por_ccla_agentid_to_portid_val (low)

Bits	Field name	Description	Type	Reset
31:0	valid	Specifies whether the Port ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

por_ccla_portfwd_ctl

Functions as the Port-to-Port forwarding control register. Works with por_ccla_portfwd_status.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD78

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63	Reserved	48	47	32
			port_fwd_req	

Figure 3-413 por_ccla_por_ccla_portfwd_ctl (high)

The following table shows the por_ccla_portfwd_ctl higher register bit assignments.

Table 3-433 por_ccla_por_ccla_portfwd_ctl (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	port_fwd_req	Bit vector, where each bit represents logical PortID of a Port (CXG) present on CMN. Each bit is used to control the communication channel with the corresponding port. 1'b0: Port-to-Port forwarding channel de-activate request 1'b1: Port-to-Port forwarding channel activate request	RW	16'b0

The following image shows the lower register bit assignments.

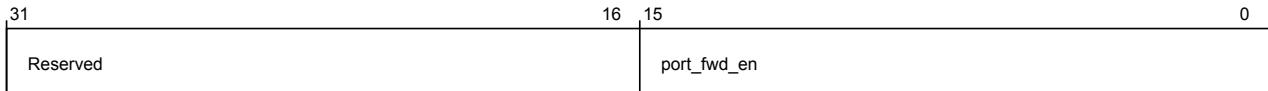


Figure 3-414 por_ccla_por_ccla_portfwd_ctl (low)

The following table shows the por_ccla_portfwd_ctl lower register bit assignments.

Table 3-434 por_ccla_por_ccla_portfwd_ctl (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	port_fwd_en	Bit vector, where each bit represents logical PortID of a Port (CXG) present on CMN. Each bit when set, enables Port-to-Port forwarding with the corresponding port. 1'b0: Port-to-Port forwarding is disabled 1'b1: Port-to-Port forwarding is enabled	RW	16'b0

por_ccla_portfwd_status

Functions as the Port-to-Port forwarding status register. Works with por_ccla_portfwd_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hD80

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-415 por_ccla_por_ccla_portfwd_status (high)

The following table shows the por_ccla_portfwd_status higher register bit assignments.

Table 3-435 por_ccla_por_ccla_portfwd_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31

16 15

0

Reserved

port_fwd_ack

Figure 3-416 por_ccla_por_ccla_portfwd_status (low)

The following table shows the por_ccla_portfwd_status lower register bit assignments.

Table 3-436 por_ccla_por_ccla_portfwd_status (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	port_fwd_ack	Bit vector, where each bit represents logical PortID of a Port (CXG) present on CMN. Each bit represents the status of the port-to-port control request sent to the corresponding port. 1'b0: Port-to-Port forwarding channel is de-active. 1'b1: Port-to-Port forwarding channel is active	RO	16'b0

por_ccla_cxl_link_rx_credit_ctl

CXL Link Rx Credit Control Register

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hE00

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

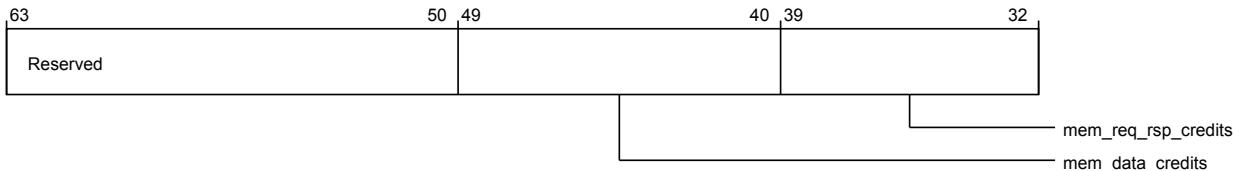


Figure 3-417 por_ccla_por_ccla_cxl_link_rx_credit_ctl (high)

The following table shows the por_ccla_cxl_link_rx_credit_ctl higher register bit assignments.

Table 3-437 por_ccla_por_ccla_cxl_link_rx_credit_ctl (high)

Bits	Field name	Description	Type	Reset
63:50	Reserved	Reserved	RO	-
49:40	mem_data_credits	Credits to advertise for Mem Data channel at init	RW	10'b0
39:32	mem_req_rsp_credits	Credits to advertise for Mem Request or Response channel at init	RW	10'b0

The following image shows the lower register bit assignments.

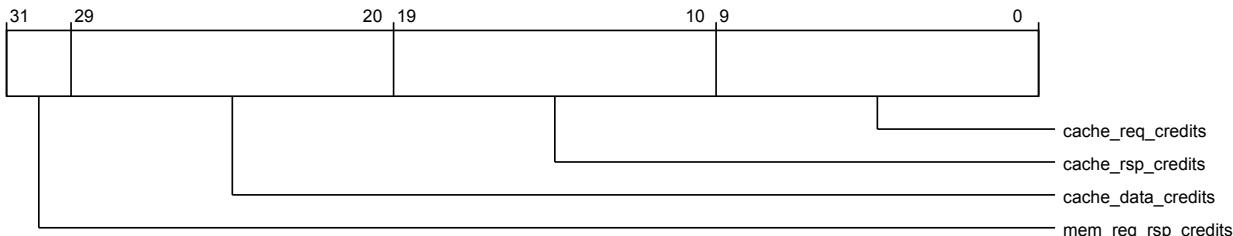


Figure 3-418 por_ccla_por_ccla_cxl_link_rx_credit_ctl (low)

The following table shows the por_ccla_cxl_link_rx_credit_ctl lower register bit assignments.

Table 3-438 por_ccla_por_ccla_cxl_link_rx_credit_ctl (low)

Bits	Field name	Description	Type	Reset
31:30	mem_req_rsp_credits	Credits to advertise for Mem Request or Response channel at init	RW	10'b0
29:20	cache_data_credits	Credits to advertise for Cache Data channel at init	RW	10'b0
19:10	cache_rsp_credits	Credits to advertise for Cache Response channel at init	RW	10'b0
9:0	cache_req_credits	Credits to advertise for Cache Request channel at init	RW	10'b0

por_ccla_cxl_link_rx_credit_return_stat

CXL Link Rx Credit Return Status Register

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hE08

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

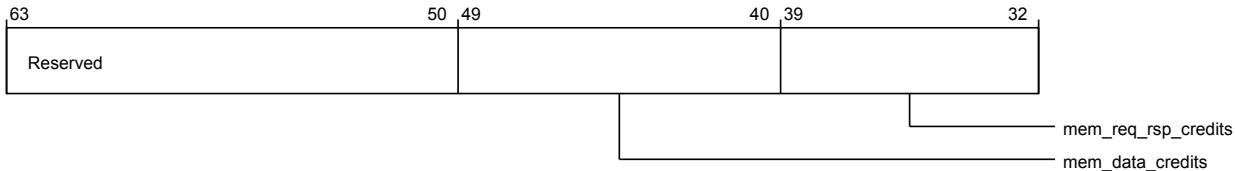


Figure 3-419 por_ccla_por_ccla_cxl_link_rx_credit_return_stat (high)

The following table shows the port class, link, rx credit, return stat higher register bit assignments.

Table 3-439 por_ccla_por_ccla_cxl_link_rx_credit_return_stat (high)

Bits	Field name	Description	Type	Reset
63:50	Reserved	Reserved	RO	-
49:40	mem_data_credits	Running snapshot of accumulated Mem Data credits to be returned	RO	10'b0
39:32	mem_req_rsp_credits	Running snapshot of accumulated Mem Request or Response credits to be returned	RO	10'b0

The following image shows the lower register bit assignments.

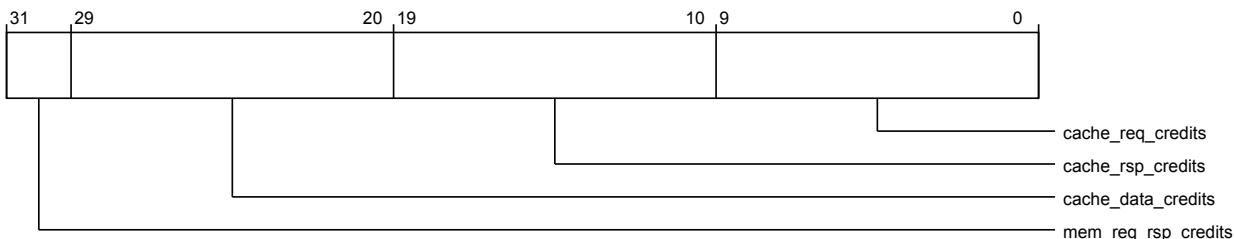


Figure 3-420 por_ccla_por_ccla_cxl_link_rx_credit_return_stat (low)

The following table shows the port class, link, rx credit, return stat, lower register bit assignments.

Table 3-440 por ccla por ccla cxl link rx credit return stat (low)

Bits	Field name	Description	Type	Reset
31:30	mem_req_rsp_credits	Running snapshot of accumulated Mem Request or Response credits to be returned	RO	10'b0
29:20	cache_data_credits	Running snapshot of accumulated Cache Data credits to be returned	RO	10'b0
19:10	cache_rsp_credits	Running snapshot of accumulated Cache Response credits to be returned	RO	10'b0
9:0	cache_req_credits	Running snapshot of accumulated Cache Request credits to be returned	RO	10'b0

por_ccla_cxl_link_tx_credit_return_stat

CXL Link Tx Credit Return Status Register

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'hE10
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

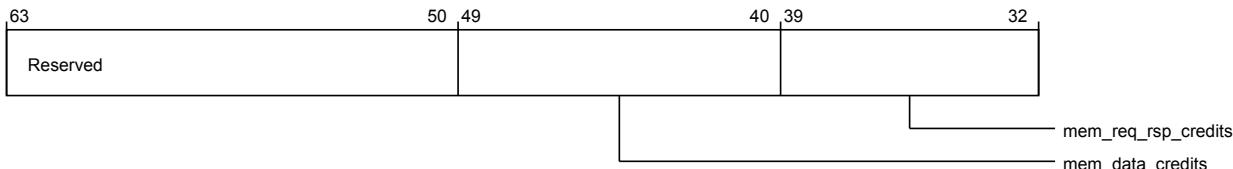


Figure 3-421 por_ccla_por_ccla_cxl_link_tx_credit_return_stat (high)

The following table shows the por_ccla_cxl_link_tx_credit_return_stat higher register bit assignments.

Table 3-441 por_ccla_por_ccla_cxl_link_tx_credit_return_stat (high)

Bits	Field name	Description	Type	Reset
63:50	Reserved	Reserved	RO	-
49:40	mem_data_credits	Running snapshot of accumulated Mem Data credits for TX	RO	10'b0
39:32	mem_req_rsp_credits	Running snapshot of accumulated Mem Request or Response credits for TX	RO	10'b0

The following image shows the lower register bit assignments.

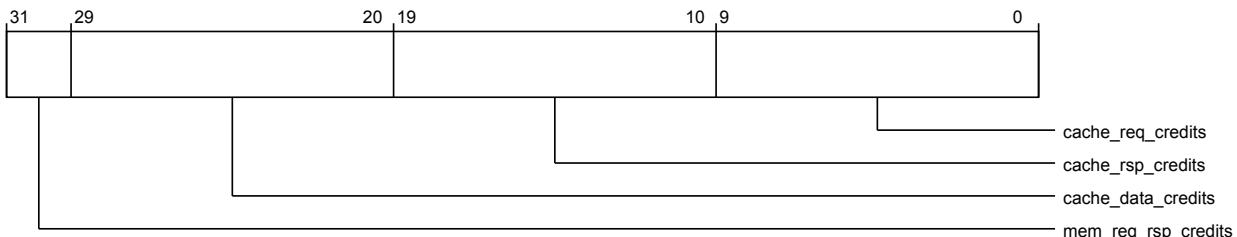


Figure 3-422 por_ccla_por_ccla_cxl_link_tx_credit_return_stat (low)

The following table shows the por_ccla_cxl_link_tx_credit_return_stat lower register bit assignments.

Table 3-442 por_ccla_por_ccla_cxl_link_tx_credit_return_stat (low)

Bits	Field name	Description	Type	Reset
31:30	mem_req_rsp_credits	Running snapshot of accumulated Mem Request or Response credits for TX	RO	10'b0
29:20	cache_data_credits	Running snapshot of accumulated Cache Data credits for TX	RO	10'b0
19:10	cache_rsp_credits	Running snapshot of accumulated Cache Response credits for TX	RO	10'b0
9:0	cache_req_credits	Running snapshot of accumulated Cache Request credits for TX	RO	10'b0

por_ccla_cxl_link_layer_defeature

CXL Link Layer Defeature Register

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hE18

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-423 por_ccla_por_ccla_cxl_link_layer_defeature (high)

The following table shows the por_ccla_cxl_link_layer_defeature higher register bit assignments.

Table 3-443 por_ccla_por_ccla_cxl_link_layer_defeature (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

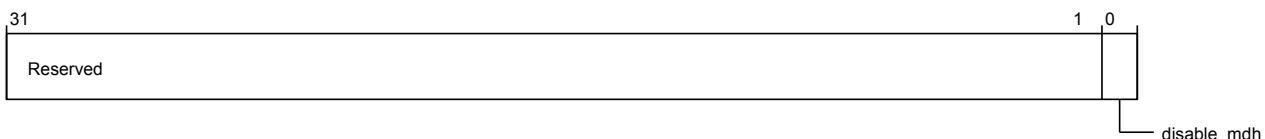


Figure 3-424 por_ccla_por_ccla_cxl_link_layer_defeature (low)

The following table shows the por_ccla_cxl_link_layer_defeature lower register bit assignments.

Table 3-444 por_ccla_por_ccla_cxl_link_layer_defeature (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	disable_mdh	Write 1 to disable MDH. Software needs to ensure it programs this value consistently on the UP and DP. After programming, a warm reset is required for the disable to take effect.	RW	1'b0

por_ccla_ull_ctl

Upper Link Layer Control Register

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE20
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

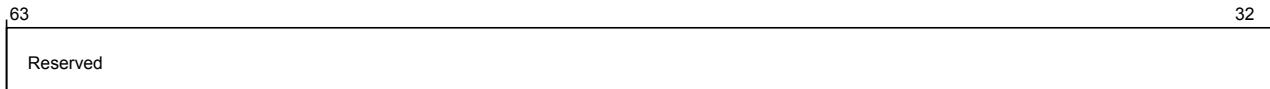


Figure 3-425 por_ccla_por_ccla_ull_ctl (high)

The following table shows the por_ccla_ull_ctl higher register bit assignments.

Table 3-445 por_ccla_por_ccla_ull_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

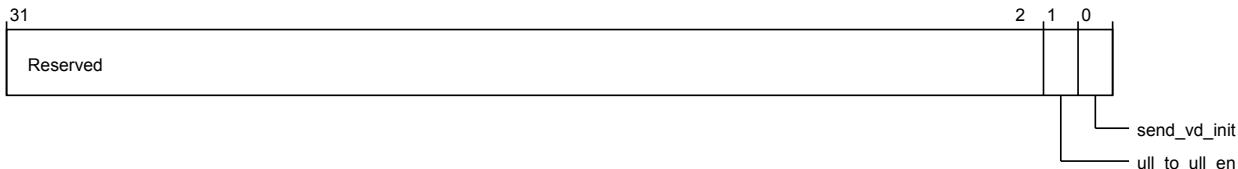


Figure 3-426 por_ccla_por_ccla_ull_ctl (low)

The following table shows the por_ccla_ull_ctl lower register bit assignments.

Table 3-446 por_ccla_por_ccla_ull_ctl (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1	ull_to_ull_en	Used to enable ULL-to_ULL mode. Must be set on both sides of the link before 'send_vd_init' is set on either side 1'b0: When clear, ull-to-ull mode is disabled. 1'b1: When set, ull-to-ull mode is enabled.	RW	1'b0
0	send_vd_init	Used to send VD Init message. Must only be used for direct ULL to ULL connection. Must be used along with Tx ULL state (tx_ull_state) status bit 1'b0: When clear and tx_ull_state is in run_state, sends VD.De-activate flit. 1'b1: When set and tx_ull_state is in stop state, sends VD.Activate flit.	RW	1'b0

por_ccla_ull_status

Upper Link Layer Status Register

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'hE28
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

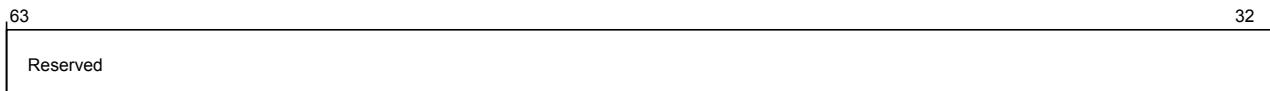


Figure 3-427 por_ccla_por_ccla_ull_status (high)

The following table shows the por_ccla_ull_status higher register bit assignments.

Table 3-447 por_ccla_por_ccla_ull_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

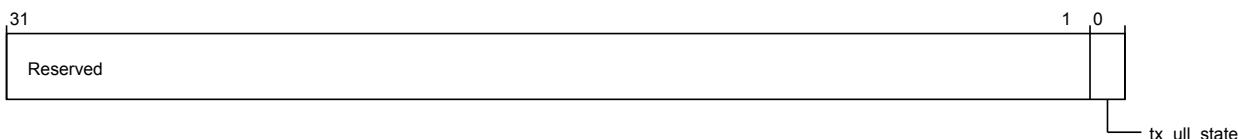


Figure 3-428 por_ccla_por_ccla_ull_status (low)

The following table shows the por_ccla_ull_status lower register bit assignments.

Table 3-448 por_ccla_por_ccla_ull_status (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	tx_ull_state	Reflects the Tx ULL state . 1'b0: Tx ULL is in Stop state 1'b1: Tx ULL is in Run state	RO	1'b0

por_ccla_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2008
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

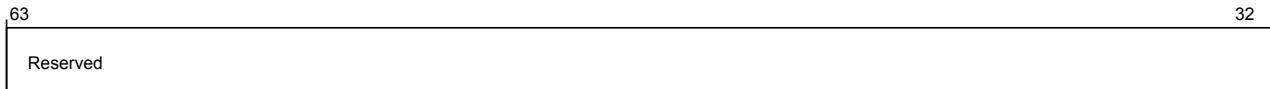


Figure 3-429 por_ccla_por_ccla_pmu_event_sel (high)

The following table shows the por_ccla_por_ccla_pmu_event_sel higher register bit assignments.

Table 3-449 por_ccla_por_ccla_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

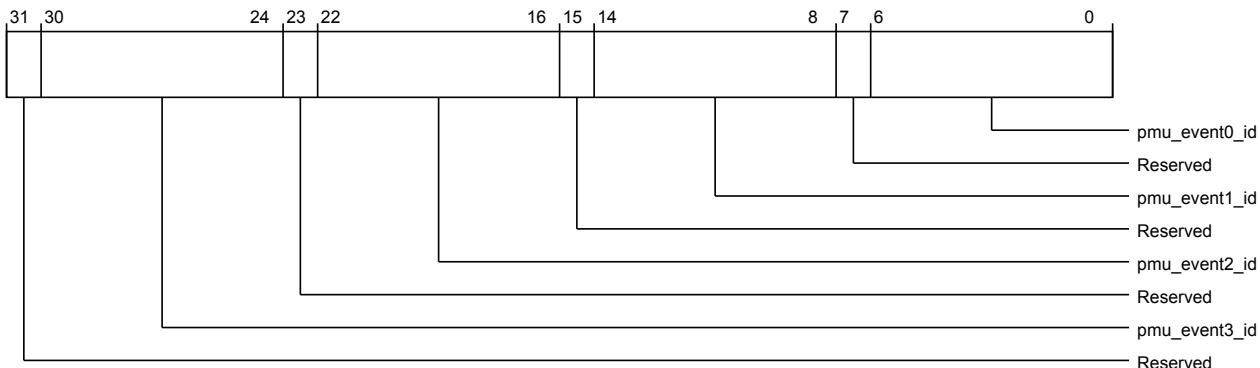


Figure 3-430 por_ccla_por_ccla_pmu_event_sel (low)

The following table shows the por_ccla_por_ccla_pmu_event_sel lower register bit assignments.

Table 3-450 por_ccla_por_ccla_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:24	pmu_event3_id	CCLA PMU Event 3 ID; see pmu_event0_id for encodings	RW	7'b0
23	Reserved	Reserved	RO	-
22:16	pmu_event2_id	CCLA PMU Event 2 ID; see pmu_event0_id for encodings	RW	7'b0
15	Reserved	Reserved	RO	-

Table 3-450 por_ccla_por_ccla_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
14:8	pmu_event1_id	CCLA PMU Event 1 ID; see pmu_event0_id for encodings	RW	7'b0
7	Reserved	Reserved	RO	-
6:0	pmu_event0_id	CCLA PMU Event 0 ID 7'h00: No event 7'h21: LA_RX_CXS : number of RX CXS beats 7'h22: LA_TX_CXS : number of TX CXS beats 7'h23: LA_RX_CXS_AVG_SIZE : average size of RX CXS beats 7'h24: LA_TX_CXS_AVG_SIZE : average size of TX CXS beats 7'h25: LA_TX_CXS_LCRD_BACKPRESSURE : CXS backpressure due to lack of CXS credits 7'h26: LA_LINK_CRDBUF_OCC : CCLA RX RAM buffer occupancy 7'h27: LA_LINK_CRDBUF_ALLOC: CCLA RX RAM buffer allocation	RW	7'b0

por_ccla_rni_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

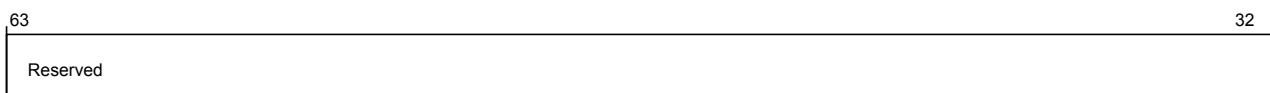


Figure 3-431 por_ccla_por_ccla_rni_secure_register_groups_override (high)

The following table shows the por_ccla_rni_secure_register_groups_override higher register bit assignments.

Table 3-451 por_ccla_por_ccla_rni_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

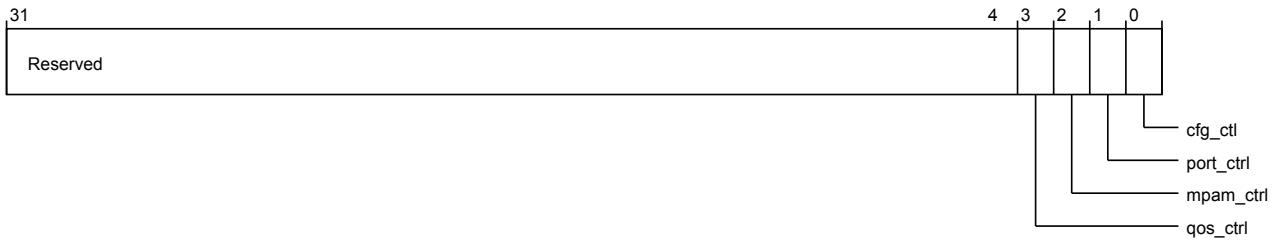


Figure 3-432 por_ccla_por_ccla_rni_secure_register_groups_override (low)

The following table shows the por_ccla_rni_secure_register_groups_override lower register bit assignments.

Table 3-452 por_ccla_por_ccla_rni_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	qos_ctrl	Allows non-secure access to secure QoS control registers	RW	1'b0
2	mpam_ctrl	Allows non-secure access to secure AXI port MPAM override register	RW	1'b0
1	port_ctrl	Allows non-secure access to secure AXI port control registers	RW	1'b0
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_ccla_rni_unit_info

Provides component identification information for RN-I.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

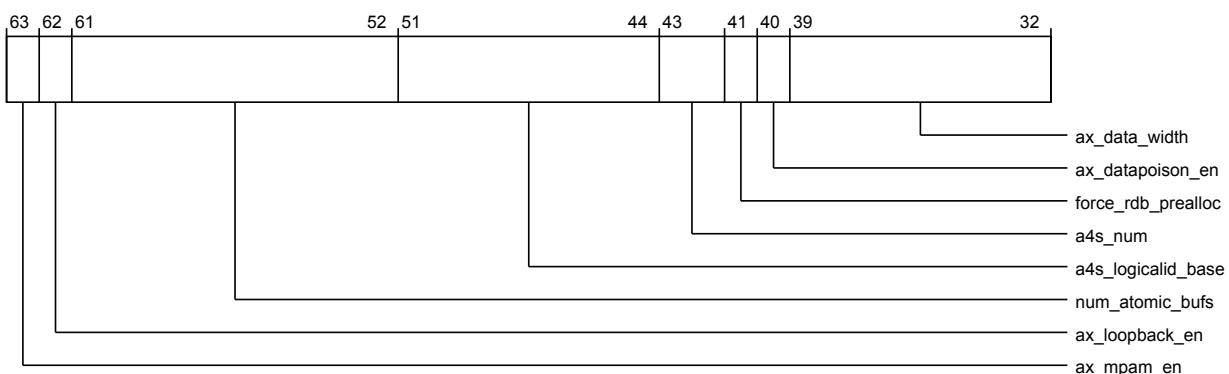


Figure 3-433 por_ccla_por_ccla_rni_unit_info (high)

The following table shows the por_ccla_rni_unit_info higher register bit assignments.

Table 3-453 por_ccla_por_ccla_rni_unit_info (high)

Bits	Field name	Description	Type	Reset
63	ax_mpam_en	MPAM enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
62	ax_loopback_en	LoopBack enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
61:52	num_atomic_bufs	Number of atomic data buffers	RO	Configuration dependent
51:44	a4s_logicalid_base	AXI4Stream interfaces logical ID base	RO	Configuration dependent
43:42	a4s_num	Number of AXI4Stream interfaces present	RO	Configuration dependent
41	force_rdb_prealloc	Force read data buffer preallocation 1'b1: yes 1'b0: no	RO	Configuration dependent
40	ax_datapoison_en	Data Poison enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
39:32	ax_data_width	AXI interface data width in bits	RO	Configuration dependent

The following image shows the lower register bit assignments.

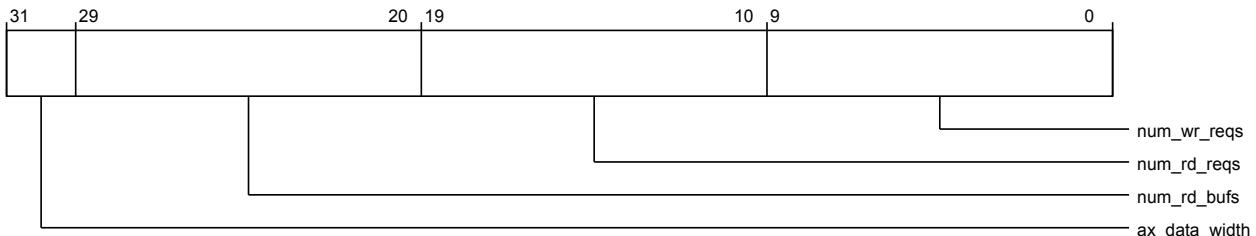


Figure 3-434 por_ccla_por_ccla_rni_unit_info (low)

The following table shows the por_ccla_rni_unit_info lower register bit assignments.

Table 3-454 por_ccla_por_ccla_rni_unit_info (low)

Bits	Field name	Description	Type	Reset
31:30	ax_data_width	AXI interface data width in bits	RO	Configuration dependent
29:20	num_rd_bufs	Number of read data buffers	RO	Configuration dependent

Table 3-454 por_ccla_por_ccla_rni_unit_info (low) (continued)

Bits	Field name	Description	Type	Reset
19:10	num_rd_reqs	Number of outstanding read requests	RO	Configuration dependent
9:0	num_wr_reqs	Number of outstanding write requests	RO	Configuration dependent

por_ccla_rni_unit_info2

Provides additional component identification information for RN-I.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h908

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-435 por_ccla_por_ccla_rni_unit_info2 (high)

The following table shows the por_ccla_rni_unit_info2 higher register bit assignments.

Table 3-455 por_ccla_por_ccla_rni_unit_info2 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

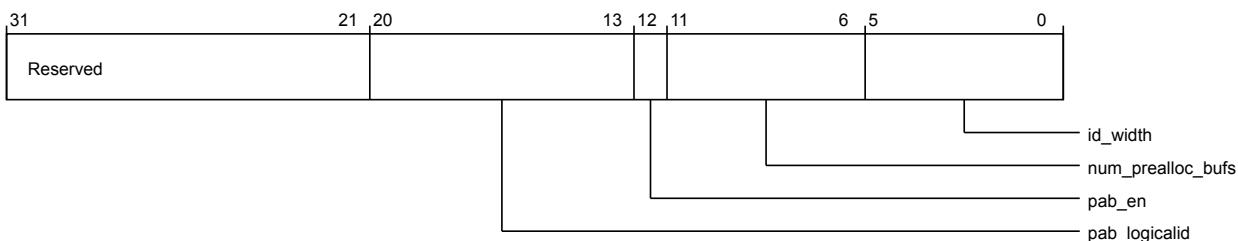


Figure 3-436 por_ccla_por_ccla_rni_unit_info2 (low)

The following table shows the por_ccla_rni_unit_info2 lower register bit assignments.

Table 3-456 por_ccla_por_ccla_rni_unit_info2 (low)

Bits	Field name	Description	Type	Reset
31:21	Reserved	Reserved	RO	-
20:13	pab_logicalid	PUB AUB bridge Logical ID	RO	Configuration dependent
12	pab_en	PUB AUB bridge enable 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
11:6	num_prealloc_bufs	Number of Pre-allocated Read Data Buffers	RO	Configuration dependent
5:0	id_width	AXI ID width for ACE-Lite slave ports	RO	Configuration dependent

por_ccla_rni_cfg_ctl

Functions as the configuration control register. Specifies the current mode.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA00

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_ccla_rni_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

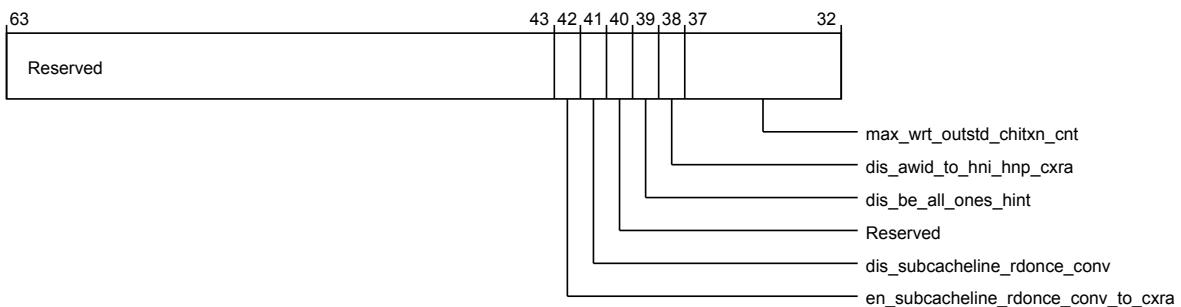


Figure 3-437 por_ccla_por_ccla_rni_cfg_ctl (high)

The following table shows the por_ccla_rni_cfg_ctl higher register bit assignments.

Table 3-457 por_ccla_por_ccla_rni_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:43	Reserved	Reserved	RO	-
42	en_subcacheline_rdonce_conv_to_cxra	If set, enables the conversion of sub-cacheline RdOnce to RdNoSnp for CXRA targets	RW	1'b0
41	dis_subcacheline_rdonce_conv	If set, disables the conversion of sub-cacheline RdOnce to RdNoSnp across all targets	RW	1'b0
40	Reserved	Reserved	RO	-
39	dis_be_all_ones_hint	If set, disables hint to HNF which signals all BE=1's on writes	RW	1'b0
38	dis_awid_to_hni_hnp_cxra	If set, disables compressed AWID to HNI and CXRA, also disables compressed AWID based ordering. Set this bit if uniq-ID write performance is needed.	RW	1'b0
37:32	max_wrt_outstd_chitxn_cnt	Maximum number of outstanding writes allowed on CHI-side	RW	Configuration dependent

The following image shows the lower register bit assignments.

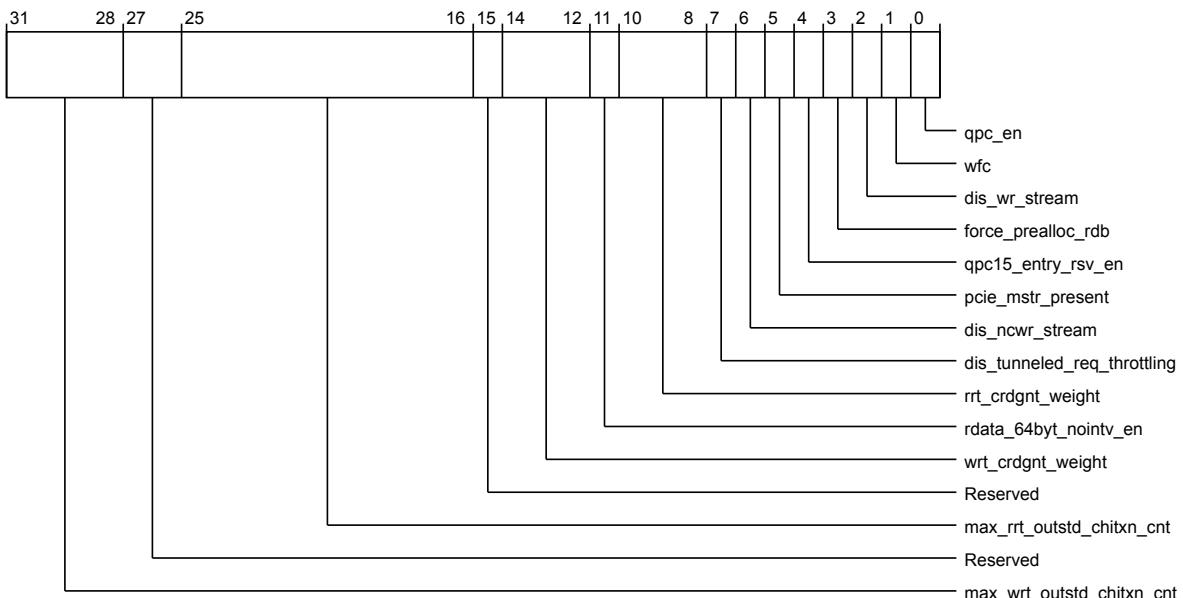


Figure 3-438 por_ccla_por_ccla_rni_cfg_ctl (low)

The following table shows the por_ccla_rni_cfg_ctl lower register bit assignments.

Table 3-458 por_ccla_por_ccla_rni_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	max_wrt_outstd_chitxn_cnt	Maximum number of outstanding writes allowed on CHI-side	RW	Configuration dependent
27:26	Reserved	Reserved	RO	-
25:16	max_rrt_outstd_chitxn_cnt	Maximum number of outstanding reads allowed on CHI-side	RW	Configuration dependent
15	Reserved	Reserved	RO	-
14:12	wrt_crdgnt_weight	Determines weight of credit grant allocated to retried writes in presence of pending retried reads	RW	3'b001
11	rdata_64byt_nointv_en	Enables no interleaving property on normal memory read data within 64B granule when set	RW	1'b1
10:8	rrt_crdgnt_weight	Determines weight of credit grant allocated to retried reads in presence of pending retried writes	RW	3'b100
7	dis_tunneled_req_throttling	Disables retry based throttling of tunneled write requests	RW	1'b0
6	dis_ncwr_stream	Disables streaming of ordered non-cacheable writes when set	RW	1'b0
5	pcie_mstr_present	Indicates PCIe master is present; must be set if PCIe master is present upstream of RN-I or RN-D	RW	1'b0
4	qpc15_entry_rsv_en	Enables QPC15 entry reservation 1'b1: Reserves tracker entry for QoS15 requests 1'b0: Does not reserve tracker entry for QoS15 requests NOTE: Only valid and applicable when por_rnd_qpc_en is set	RW	1'b0
3	force_prealloc_rdb	When set, all reads from the RN-I are sent with a preallocated read data buffer	RW	Configuration dependent
2	dis_wr_stream	Disables streaming of ordered writes when set	RW	1'b0
1	wfc	When set, enables waiting for completion (COMP) before dispatching dependent transaction (TXN)	RW	1'b0
0	qpc_en	When set, enables QPC-based scheduling using two QoS priority classes (QoS15 and non-QoS15)	RW	1'b1

por_ccla_rni_aux_ctl

Functions as the auxiliary control register for RN-I.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	Configuration dependent

Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
--------------------------	--

The following image shows the higher register bit assignments.

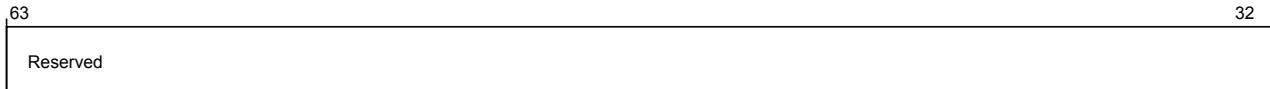


Figure 3-439 por_ccla_por_ccla_rni_aux_ctl (high)

The following table shows the por_ccla_rni_aux_ctl higher register bit assignments.

Table 3-459 por_ccla_por_ccla_rni_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

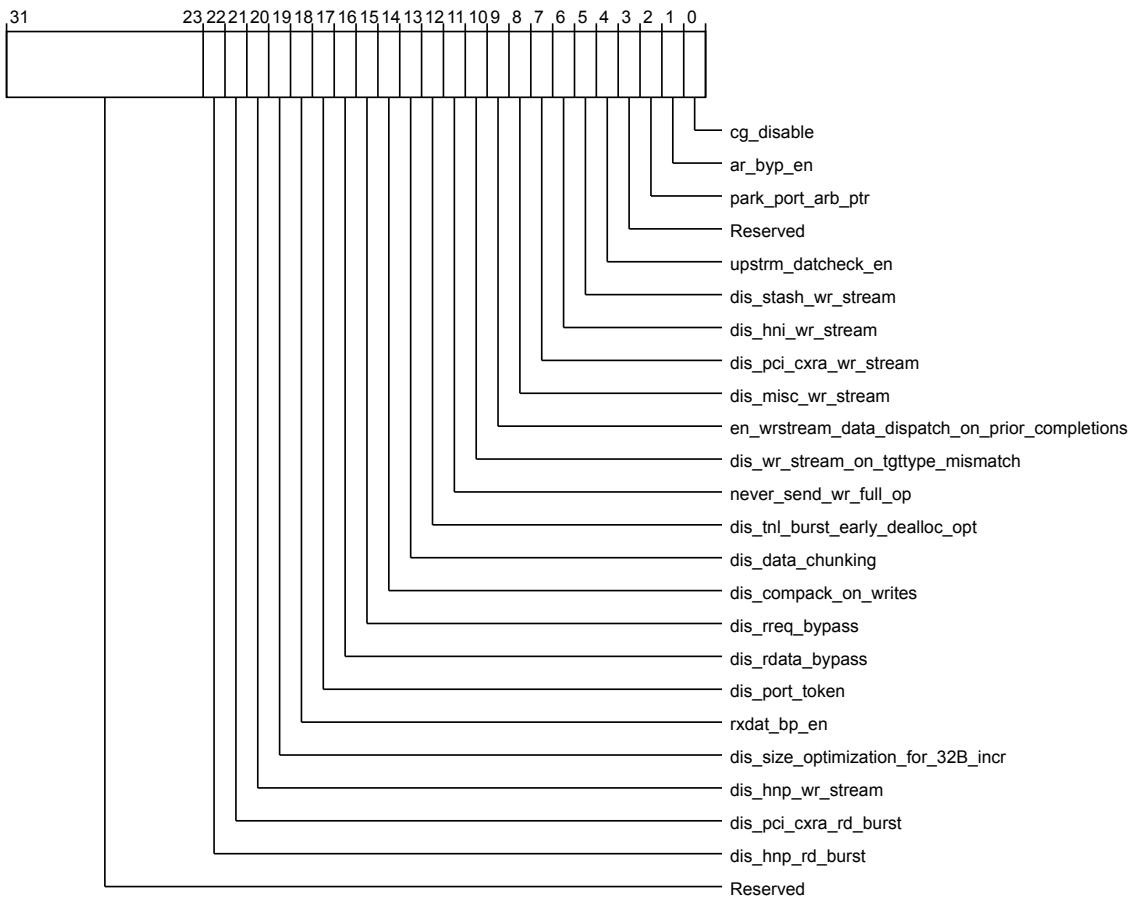


Figure 3-440 por_ccla_por_ccla_rni_aux_ctl (low)

The following table shows the por_ccla_rni_aux_ctl lower register bit assignments.

Table 3-460 por_ccla_por_ccla_rni_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:23	Reserved	Reserved	RO	-
22	dis_hnp_rd_burst	If set, disables read burst to HNP on CHI request flits . Read burst on CHI is supported only in non-decoupled RDB configuration.	RW	1'b0
21	dis_pci_cxra_rd_burst	If set, disables read burst to PCI-CXRA on CHI request flits . Read burst on CHI is supported only in non-decoupled RDB configuration.	RW	1'b0
20	dis_hnp_wr_stream	Disables streaming of ordered writes to HNP when set	RW	1'b0
19	dis_size_optimization_for_32B_incr	If set, disables the size related optimization for a 32B INCR burst (rh-2512)	RW	1'b0
18	rxdat_bp_en	If set, back pressures the rxdat interface when RDB's are not available	RW	1'b0
17	dis_port_token	If set, disables per port reservation in the tracker(rd and wr)	RW	1'b1
16	dis_rdata_bypass	If set, disables read data bypass path	RW	1'b0
15	dis_rreq_bypass	If set, disables read request bypass path	RW	1'b0
14	dis_compack_on_writes	If set, disables comp_ack on streaming writes. WrData is used for ordering writes	RW	1'b1
13	dis_data_chunking	If set, disables the data chunking feature	RW	1'b0
12	dis_tnl_burst_early_dealloc_opt	If set, disables the optimization related to early deallocation of tunnelled writes for intermediate txns of burst	RW	1'b0
11	never_send_wr_full_op	If set, RNI will never send WR FULL op. All write ops will be of PTL type	RW	1'b0
10	dis_wr_stream_on_tgtype_mismatch	If set, serializes first write when moving from one tgtype to another	RW	1'b0
9	en_wrstream_data_dispatch_on_prior_completions	If set, data dispatch for streaming writes waits for completion of all older writes	RW	1'b0
8	dis_misc_wr_stream	Disables streaming of ordered writes with following attributes when set : Device memory or EWA=0 or Excl=1	RW	1'b0
7	dis_pci_cxra_wr_stream	Disables streaming of ordered writes to PCI-CXRA when set	RW	1'b0

Table 3-460 por_ccla_por_ccla_rni_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
6	dis_hni_wr_stream	Disables streaming of ordered writes to HNI when set	RW	1'b0
5	dis_stash_wr_stream	Disables streaming of ordered WrUniqStash when set	RW	1'b0
4	upstrm_datcheck_en	Upstream supports Datacheck	RW	Configuration dependent
3	Reserved	Reserved	RO	-
2	park_port_arb_ptr	Parks the AXI port arbitration pointer for Burst	RW	1'b0
1	ar_byp_en	AR bypass enable; enables bypass path in the AR pipeline	RW	1'b1
0	cg_disable	Disables clock gating when set	RW	1'b0

por_ccla_rni_s_0-2_port_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls port S#{index} AXI/ACE slave interface settings.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA10 + # {[0, 1, 2]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccla_rni_secure_register_groups_override.port_ctrl

The following image shows the higher register bit assignments.

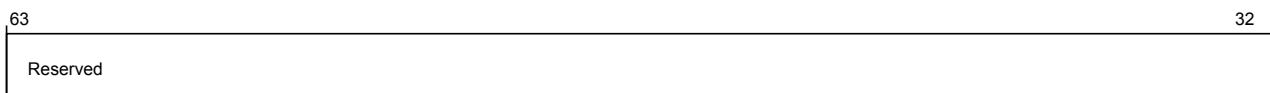


Figure 3-441 por_ccla_por_ccla_rni_s_0-2_port_control (high)

The following table shows the por_ccla_rni_s_0-2_port_control higher register bit assignments.

Table 3-461 por_ccla_por_ccla_rni_s_0-2_port_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

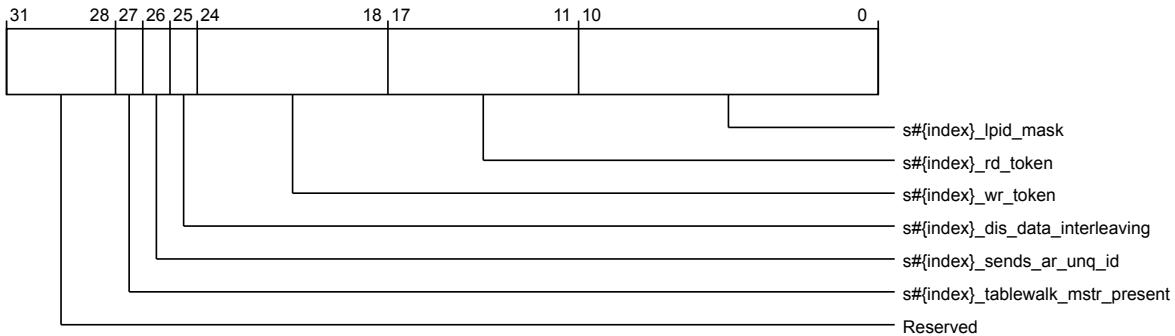


Figure 3-442 por_ccla_por_ccla_rni_s_0-2_port_control (low)

The following table shows the por_ccla_rni_s_0-2_port_control lower register bit assignments.

Table 3-462 por_ccla_por_ccla_rni_s_0-2_port_control (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27	s#{index}_tablewalk_mstr_present	If set, Indicates translation table walk master present such as TCU or GIC	RW	1'b0
26	s#{index}_sends_ar_unq_id	If set, indicates AR transactions on Port#{index} are always Unique ID. This bit for a port must be set to 1 to enable Read Burst on the CHI side of RND. This bit works in conjunction with dis_rd_burst being 0.	RW	1'b0
25	s#{index}_dis_data_interleaving	If set, disables read DATA interleaving on RDATAS#{index} channel. This applies only to RDATA generated as a response to requests on AR channel . This does not apply to RDATA generated as a response to Atomic request on AW channel. I.e. RDATA of an Atomic op, on AW channel, may interleave with RDATA of an AR channel request	RW	1'b0
24:18	s#{index}_wr_token	Port S#{index} reserved token count for AW channel This must be less than the number of Wr requests(RNID_NUM_XRT_REQ) on AW achnnel	RW	6'b00_0000

Table 3-462 por_ccla_por_ccla_rni_s_0-2_port_control (low) (continued)

Bits	Field name	Description	Type	Reset
17:11	s#{index}_rd_token	Port S#{index} reserved token count for AR channel per slice This should be less than the number of Rd requests(RNID_NUM_XRT_SLICE_REQ) per slice on AR achnnel	RW	6'b00_0000
10:0	s#{index}_lpid_mask	Port S#{index} LPID mask LPID[0]: Equal to the result of UnaryOR of BitwiseAND of LPID mask and AXID (LPID[0] = (AXID and mask)); specifies which AXID bit is reflected in the LSB of LPID LPID[2:1]: Equal to port ID[1:0]; the MSB of LPID contains port ID	RW	11'b000_0000_0000

por_ccla_rni_s_0-2_mpam_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls port S#{index} AXI/ACE slave interface MPAM override values

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA28 + # {[0, 1, 2]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccla_rni_secure_register_groups_override.mpam_ctrl

The following image shows the higher register bit assignments.



Figure 3-443 por_ccla_por_ccla_rni_s_0-2_mpam_control (high)

The following table shows the por_ccla_rni_s_0-2_mpam_control higher register bit assignments.

Table 3-463 por_ccla_por_ccla_rni_s_0-2_mpam_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

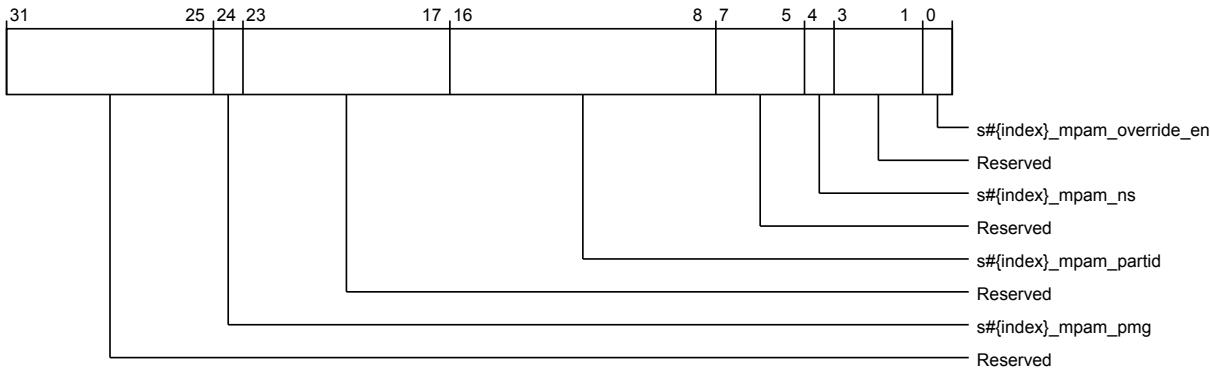


Figure 3-444 por_ccla_por_ccla_rni_s_0-2_mpam_control (low)

The following table shows the por_ccla_rni_s_0-2_mpam_control lower register bit assignments.

Table 3-464 por_ccla_por_ccla_rni_s_0-2_mpam_control (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	s#{index}_mpam_pmg	Port S#{index} MPAM_PMG value	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	s#{index}_mpam_partid	Port S#{index} MPAM_PARTID value	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	s#{index}_mpam_ns	Port S#{index} MPAM_NS value	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	s#{index}_mpam_override_en	Port S#{index} MPAM override en When set, MPAM value on CHI side is driven from MPAM override value in this register. Note that when RNID_AXMPAM_EN_PARAM is set to 0, MPAM override value is always used irrespective of this bit value	RW	1'b0

por_ccla_rni_s_0-2_qos_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls QoS settings for port S#{index} AXI/ACE slave interface.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA80 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccla_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

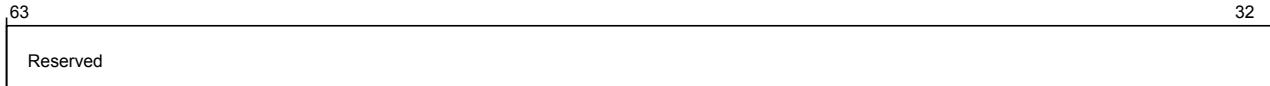


Figure 3-445 por_ccla_por_ccla_rni_s_0-2_qos_control (high)

The following table shows the por_ccla_mi_s_0-2_qos_control higher register bit assignments.

Table 3-465 por_ccla_por_ccla_rni_s_0-2_qos_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

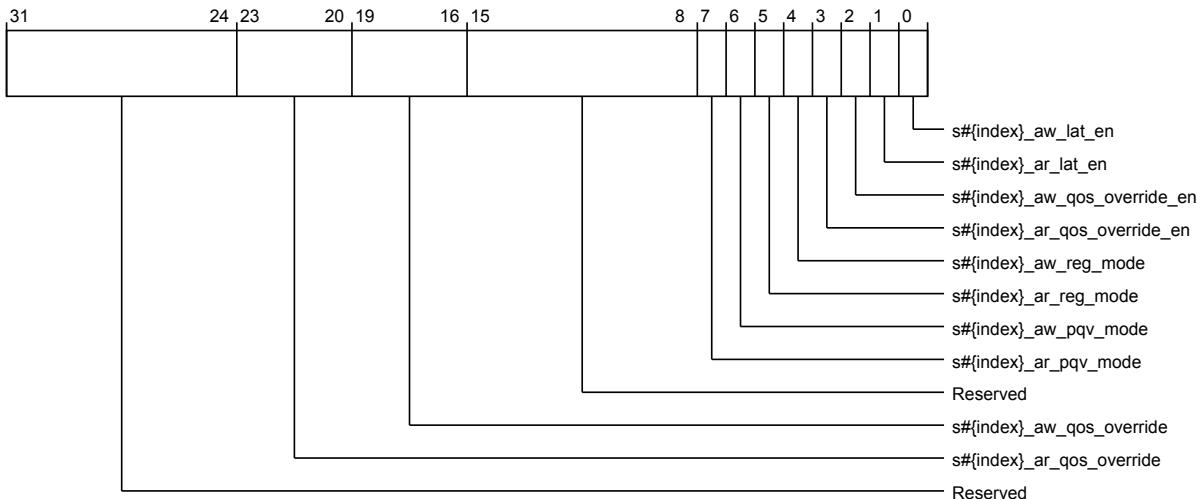


Figure 3-446 por_ccla_por_ccla_rni_s_0-2_qos_control (low)

The following table shows the por_ccla_mi_s_0-2_qos_control lower register bit assignments.

Table 3-466 por_ccla_por_ccla_rni_s_0-2_qos_control (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:20	s#{index}_ar_qos_override	AR QoS override value for port S#{index}	RW	4'b0000
19:16	s#{index}_aw_qos_override	AW QoS override value for port S#{index}	RW	4'b0000
15:8	Reserved	Reserved	RO	-

Table 3-466 por_ccla_por_ccla_rni_s_0-2_qos_control (low) (continued)

Bits	Field name	Description	Type	Reset
7	s#{index}_ar_pqv_mode	Configures the QoS regulator mode for read transactions during period mode 1'b0: Normal mode; QoS value is stable when the master is idle 1'b1: Quiesce high mode; QoS value tends to the maximum value when the master is idle	RW	1'b0
6	s#{index}_aw_pqv_mode	Configures the QoS regulator mode for write transactions during period mode 1'b0: Normal mode; QoS value is stable when the master is idle 1'b1: Quiesce high mode; QoS value tends to the maximum value when the master is idle	RW	1'b0
5	s#{index}_ar_reg_mode	Configures the QoS regulator mode for read transactions 1'b0: Latency mode 1'b1: Period mode; used for bandwidth regulation	RW	1'b0
4	s#{index}_aw_reg_mode	Configures the QoS regulator mode for write transactions 1'b0: Latency mode 1'b1: Period mode; used for bandwidth regulation	RW	1'b0
3	s#{index}_ar_qos_override_en	Enables port S#{index} AR QoS override; when set, allows QoS value on inbound AR transactions to be overridden	RW	1'b0
2	s#{index}_aw_qos_override_en	Enables port S#{index} AW QoS override; when set, allows QoS value on inbound AW transactions to be overridden	RW	1'b0
1	s#{index}_ar_lat_en	Enables port S#{index} AR QoS regulation when set	RW	1'b0
0	s#{index}_aw_lat_en	Enables port S#{index} AW QoS regulation when set	RW	1'b0

por_ccla_rni_s_0-2_qos_lat_tgt

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls QoS target latency (in cycles) for regulations of port S#{index} read and write transactions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA88 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccla_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

63 Reserved 32

Figure 3-447 por_ccla_por_ccla_rni_s_0-2_qos_lat_tgt (high)

The following table shows the por_ccla_rni_s_0-2_qos_lat_tgt higher register bit assignments.

Table 3-467 por_ccla_por_ccla_rni_s_0-2_qos_lat_tgt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

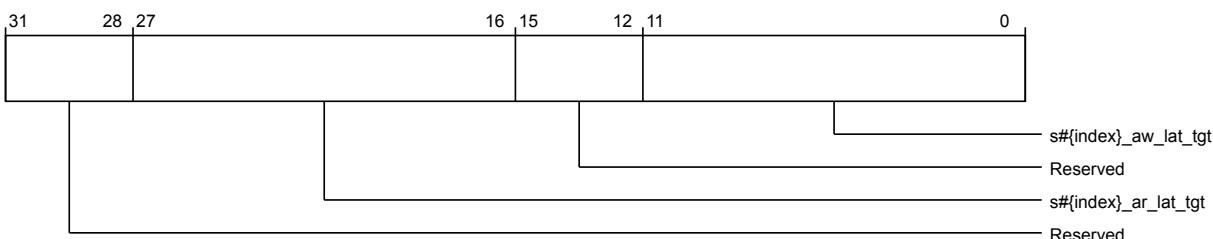


Figure 3-448 por_ccla_por_ccla_rni_s_0-2_qos_lat_tgt (low)

The following table shows the port class assignments for QoS latency target lower register bit assignments.

Table 3-468 por ccla por ccla rni s 0-2 qos lat tgt (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:16	s#{index}_ar_lat_tgt	Port S#{index} AR channel target latency; a value of 0 corresponds to no regulation	RW	12'h000
15:12	Reserved	Reserved	RO	-
11:0	s#{index}_aw_lat_tgt	Port S#{index} AW channel target latency; a value of 0 corresponds to no regulation	RW	12'h000

por_ccla_rni_s_0-2_qos_lat_scale

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls the QoS target latency scale factor for port S#{index} read and write transactions. This register represents powers of two from the range $2^{(-5)}$ to $2^{(-12)}$; it is used to match a 16-bit integrator.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA

Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the

Secure group override por_ccla_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

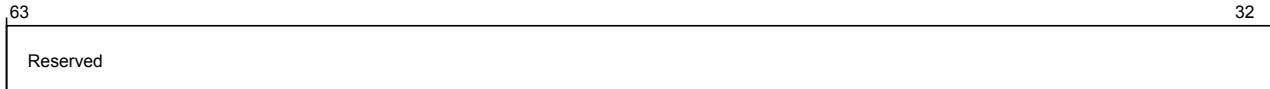


Figure 3-449 por_ccla_por_ccla_rni_s_0-2_qos_lat_scale (high)

The following table shows the por_ccla_rni_s_0-2_qos_lat_scale higher register bit assignments.

Table 3-469 por_ccla_por_ccla_rni_s_0-2_qos_lat_scale (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

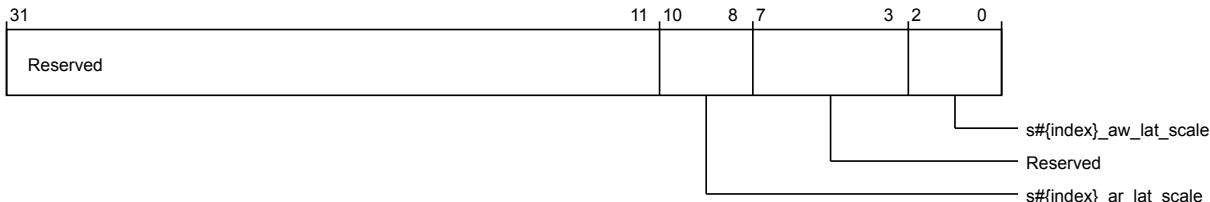


Figure 3-450 por_ccla_por_ccla_rni_s_0-2_qos_lat_scale (low)

The following table shows the por_ccla_rni_s_0-2_qos_lat_scale lower register bit assignments.

Table 3-470 por_ccla_por_ccla_rni_s_0-2_qos_lat_scale (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10:8	s#{index}_ar_lat_scale	Port S#{index} AR QoS scale factor 3'b000: 2^(-5) 3'b001: 2^(-6) 3'b010: 2^(-7) 3'b011: 2^(-8) 3'b100: 2^(-9) 3'b101: 2^(-10) 3'b110: 2^(-11) 3'b111: 2^(-12)	RW	3'h0

Table 3-470 por_ccla_por_ccla_rni_s_0-2_qos_lat_scale (low) (continued)

Bits	Field name	Description	Type	Reset
7:3	Reserved	Reserved	RO	-
2:0	s#{index}_aw_lat_scale	Port S#{index} AW QoS scale factor 3'b000: 2 ⁽⁻⁵⁾ 3'b001: 2 ⁽⁻⁶⁾ 3'b010: 2 ⁽⁻⁷⁾ 3'b011: 2 ⁽⁻⁸⁾ 3'b100: 2 ⁽⁻⁹⁾ 3'b101: 2 ⁽⁻¹⁰⁾ 3'b110: 2 ⁽⁻¹¹⁾ 3'b111: 2 ⁽⁻¹²⁾	RW	3'h0

por_ccla_rni_s_0-2_qos_lat_range

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls the minimum and maximum QoS values generated by the QoS latency regulator for port S#{index} read and write transactions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA98 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccla_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

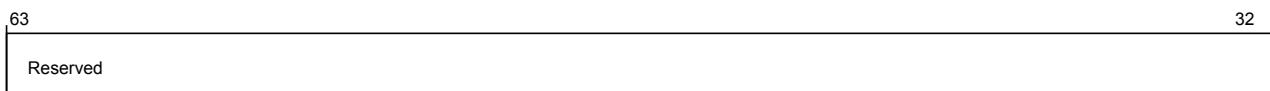


Figure 3-451 por_ccla_por_ccla_rni_s_0-2_qos_lat_range (high)

The following table shows the por_ccla_rni_s_0-2_qos_lat_range higher register bit assignments.

Table 3-471 por_ccla_por_ccla_rni_s_0-2_qos_lat_range (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

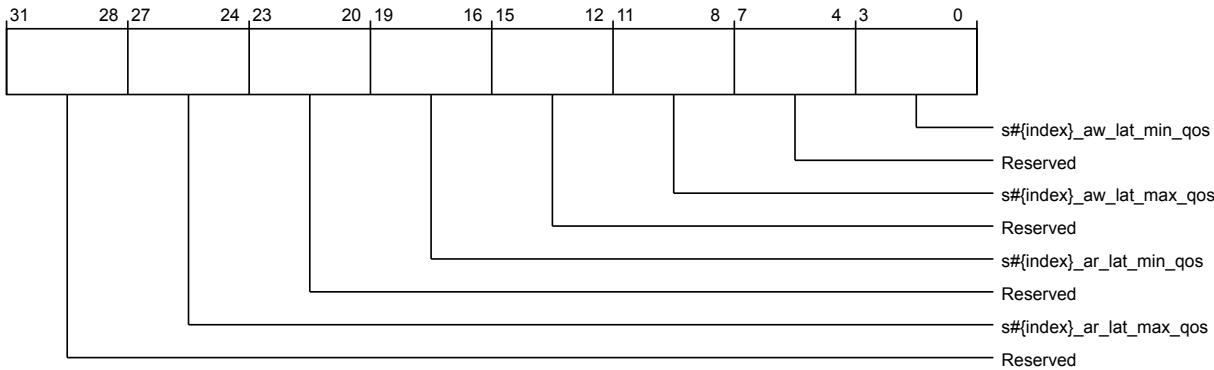


Figure 3-452 por_ccla_por_ccla_rni_s_0-2_qos_lat_range (low)

The following table shows the por_ccla_rni_s_0-2_qos_lat_range lower register bit assignments.

Table 3-472 por_ccla_por_ccla_rni_s_0-2_qos_lat_range (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	s#{index}_ar_lat_max_qos	Port S#{index} AR QoS maximum value	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	s#{index}_ar_lat_min_qos	Port S#{index} AR QoS minimum value	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	s#{index}_aw_lat_max_qos	Port S#{index} AW QoS maximum value	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	s#{index}_aw_lat_min_qos	Port S#{index} AW QoS minimum value	RW	4'h0

por_ccla_rni_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

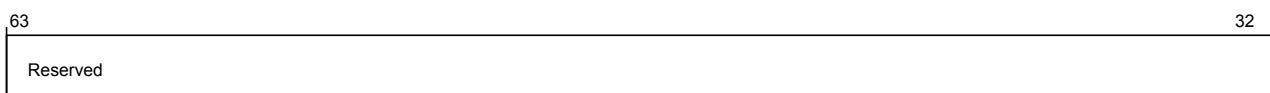


Figure 3-453 por_ccla_por_ccla_rni_pmu_event_sel (high)

The following table shows the por_ccla_rni_pmu_event_sel higher register bit assignments.

Table 3-473 por_ccla_por_ccla_rni_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

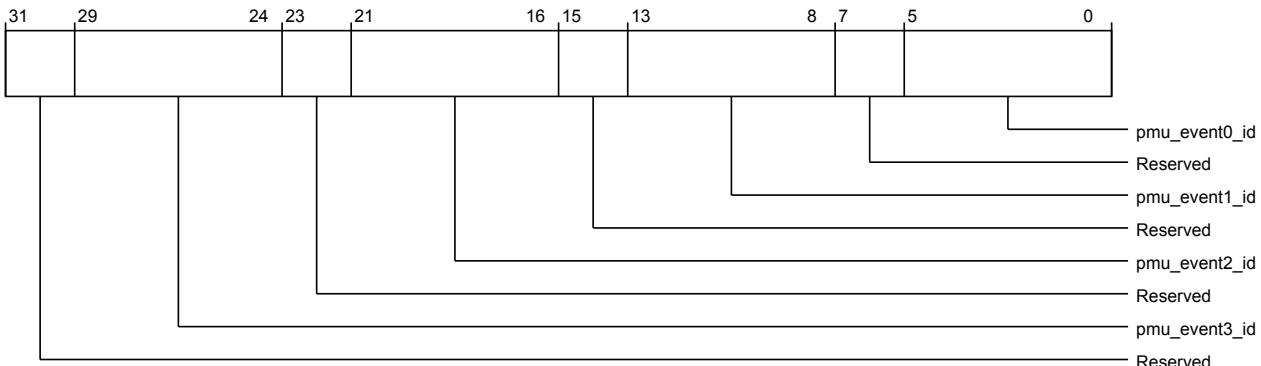


Figure 3-454 por_ccla_por_ccla_rni_pmu_event_sel (low)

The following table shows the port class name pmu event selector lower register bit assignments.

Table 3-474 por_ccla_por_ccla_rni_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	RN-I PMU Event 3 ID; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	RN-I PMU Event 2 ID; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	RN-I PMU Event 1 ID; see pmu_event0_id for encodings	RW	6'b0

Table 3-474 por_ccla_por_ccla_rni_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-

Table 3-474 por_ccla_por_ccla_rni_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
5:0	pmu_event0_id	RN-I PMU Event 0 ID 6'h00: No event 6'h01: Port S0 RDataBeats 6'h02: Port S1 RDataBeats 6'h03: Port S2 RDataBeats 6'h04: RXDAT flits received 6'h05: TXDAT flits sent 6'h06: Total TXREQ flits sent 6'h07: Retried TXREQ flits sent 6'h08: RRT occupancy count overflow 6'h09: WRT occupancy count overflow 6'h0A: Replayed TXREQ flits 6'h0B: WriteCancel sent 6'h0C: Port S0 WDataBeats 6'h0D: Port S1 WDataBeats 6'h0E: Port S2 WDataBeats 6'h0F: RRT allocation 6'h10: WRT allocation 6'h11: PADB occupancy count overflow 6'h12: RPDB occupancy count overflow 6'h13: RRT occupancy count overflow_slice1 6'h14: RRT occupancy count overflow_slice2 6'h15: RRT occupancy count overflow_slice3 6'h16: WRT request throttled 6'h17: RNI backpressure CHI LDB full 6'h18: RRT normal rd req occupancy count overflow_slice0 6'h19: RRT normal rd req occupancy count overflow_slice1 6'h1A: RRT normal rd req occupancy count overflow_slice2 6'h1B: RRT normal rd req occupancy count overflow_slice3 6'h1C: RRT PCIe RD burst req occupancy count overflow_slice0 6'h1D: RRT PCIe RD burst req occupancy count overflow_slice1 6'h1E: RRT PCIe RD burst req occupancy count overflow_slice2 6'h1F: RRT PCIe RD burst req occupancy count overflow_slice3 6'h20: RRT PCIe RD burst allocation 6'h21: Compressed AWID ordering 6'h22: Atomic data buffer allocation 6'h23: Atomic data buffer occupancy	RW	6'b0

3.3.9 CXHA register descriptions

This section lists the CXHA registers.

por_cxg_ha_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

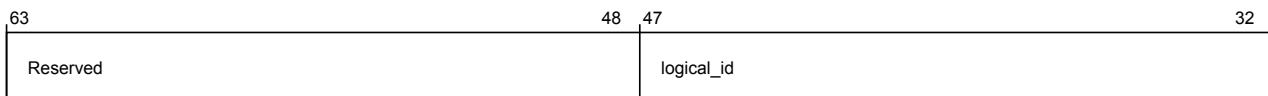


Figure 3-455 por_cxg_ha_por_cxg_ha_node_info (high)

The following table shows the por_cxg_ha_node_info higher register bit assignments.

Table 3-475 por_cxg_ha_por_cxg_ha_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

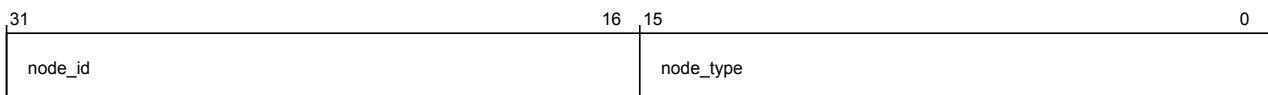


Figure 3-456 por_cxg_ha_por_cxg_ha_node_info (low)

The following table shows the por_cxg_ha_node_info lower register bit assignments.

Table 3-476 por_cxg_ha_por_cxg_ha_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0101

por_cxg_ha_id

Contains the CCIX-assigned HAID.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-457 por_cxg_ha_por_cxg_ha_id (high)

The following table shows the por_cxg_ha_id higher register bit assignments.

Table 3-477 por_cxg_ha_por_cxg_ha_id (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

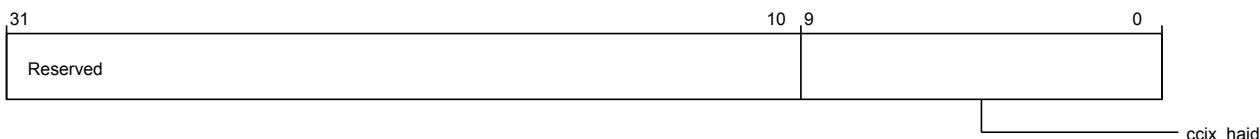


Figure 3-458 por_cxg_ha_por_cxg_ha_id (low)

The following table shows the por_cxg_ha_id lower register bit assignments.

Table 3-478 por_cxg_ha_por_cxg_ha_id (low)

Bits	Field name	Description	Type	Reset
31:10	Reserved	Reserved	RO	-
9:0	ccix_haid	CCIX HAID	RW	10'h0

por_cxg_ha_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset	16'h80
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-459 por_cxg_ha_por_cxg_ha_child_info (high)

The following table shows the por_cxg_ha_child_info higher register bit assignments.

Table 3-479 por_cxg_ha_por_cxg_ha_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

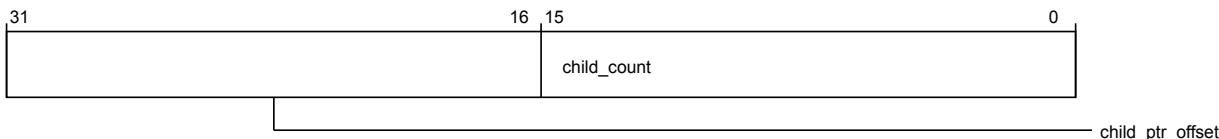


Figure 3-460 por_cxg_ha_por_cxg_ha_child_info (low)

The following table shows the por_cxg_ha_child_info lower register bit assignments.

Table 3-480 por_cxg_ha_por_cxg_ha_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'h0

por_cxg_ha_cfg_ctl

Functions as the configuration control register. Specifies the current mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_cxg_ha_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

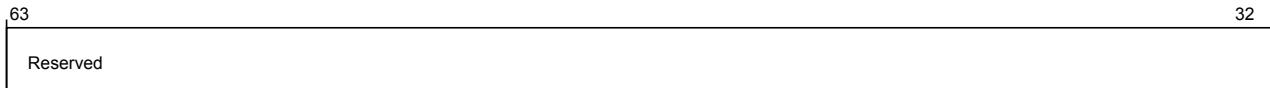


Figure 3-461 por_cxg_ha_por_cxg_ha_cfg_ctl (high)

The following table shows the por_cxg_ha_cfg_ctl higher register bit assignments.

Table 3-481 por_cxg_ha_por_cxg_ha_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

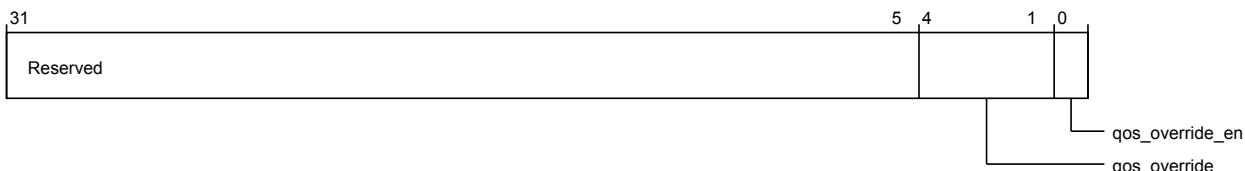


Figure 3-462 por_cxg_ha_por_cxg_ha_cfg_ctl (low)

The following table shows the por_cxg_ha_cfg_ctl lower register bit assignments.

Table 3-482 por_cxg_ha_por_cxg_ha_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:5	Reserved	Reserved	RO	-
4:1	qos_override	QoS override value	RW	4'b0
0	qos_override_en	QoS override en When set, QoS value on CHI side is driven from QoS override value in this register	RW	1'b0

por_cxg_ha_aux_ctl

Functions as the auxiliary control register for CXHA.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA08

Register reset 64'b000100000001000

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-463 por_cxg_ha_por_cxg_ha_aux_ctl (high)

The following table shows the por_cxg_ha_aux_ctl higher register bit assignments.

Table 3-483 por_cxg_ha_por_cxg_ha_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

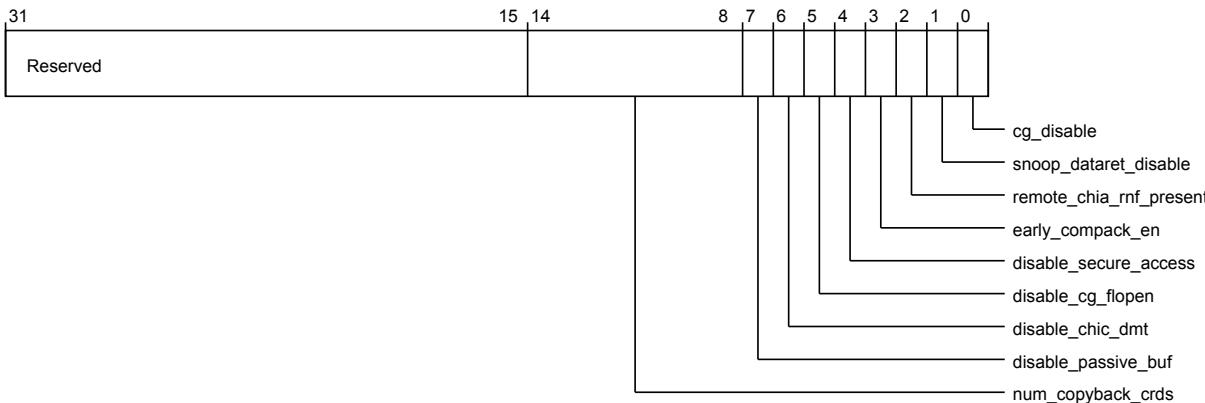


Figure 3-464 por_cxg_ha_por_cxg_ha_aux_ctl (low)

The following table shows the por_cxg_ha_aux_ctl lower register bit assignments.

Table 3-484 por_cxg_ha_por_cxg_ha_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:8	num_copyback_crds	Controls the total number of Request/Data credits reserved for CopyBack Requests across all links Note: This should be less than the Write Data Buffer depth	RW	7'h40
7	disable_passive_buf	When set disables Passive Buffer	RW	1'b0
6	disable_chic_dmt	When set disables CHI-C style DMT	RW	1'b0
5	disable_cg_flopen	Disables enhanced flop enable control for dynamic power savings	RW	1'b0
4	disable_secure_access	Converts all accesses to non-secure	RW	1'b0
3	early_compack_en	Early CompAck enable; enables sending early CompAck on CCIX for requests that require CompAck	RW	1'b1

Table 3-484 por_cxg_ha_por_cxg_ha_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	remote_chia_rnf_present	Indicates existence of CHIA RN-F in system; HA uses this indication to send SnpToS or SnpToSC 1'b0: HA converts SnpShared, SnpClean, and SnpNotSharedDirty to SnpToSC 1'b1: HA converts SnpShared, SnpClean, and SnpNotSharedDirty to SnpToS	RW	1'b0
1	snoop_dataret_disable	Disables setting data return for CCIX snoop requests for all CHI snoop opcodes	RW	1'b0
0	cg_disable	Disables clock gating when set	RW	1'b0

por_cxg_ha_mpam_control

Controls MPAM override values on incoming CCIX Request in non-SMP mode

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA10
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_cxg_ha_secure_register_groups_override.mpam_ctl

The following image shows the higher register bit assignments.

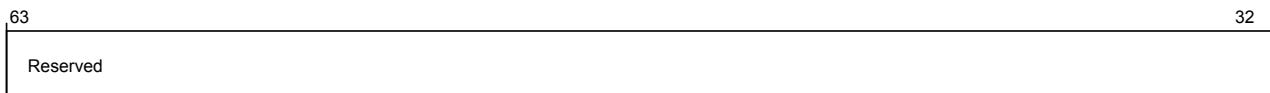


Figure 3-465 por_cxg_ha_por_cxg_ha_mpam_control (high)

The following table shows the por_cxg_ha_mpam_control higher register bit assignments.

Table 3-485 por_cxg_ha_por_cxg_ha_mpam_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

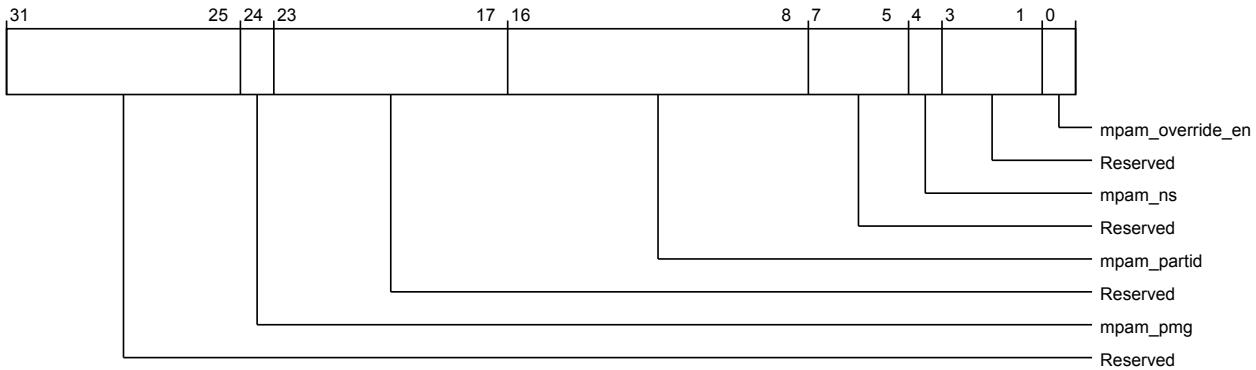


Figure 3-466 por_cxg_ha_por_cxg_ha_mpam_control (low)

The following table shows the por_cxg_ha_mpam_control lower register bit assignments.

Table 3-486 por_cxg_ha_por_cxg_ha_mpam_control (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	mpam_pmg	MPAM_PMG value	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	mpam_partid	MPAM_PARTID value	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	mpam_ns	MPAM_NS value	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	mpam_override_en	MPAM override en When set, MPAM value on CHI side is driven from MPAM override value in this register. Applicable only in non-SMP mode	RW	1'b0

por_cxg_ha_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

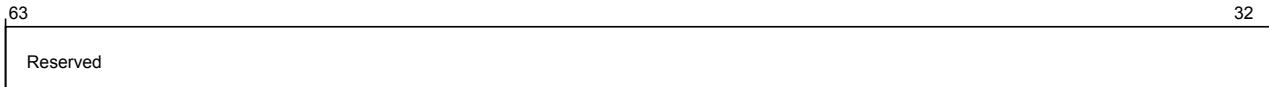


Figure 3-467 por_cxg_ha_por_cxg_ha_secure_register_groups_override (high)

The following table shows the por_cxg_ha_secure_register_groups_override higher register bit assignments.

Table 3-487 por_cxg_ha_por_cxg_ha_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

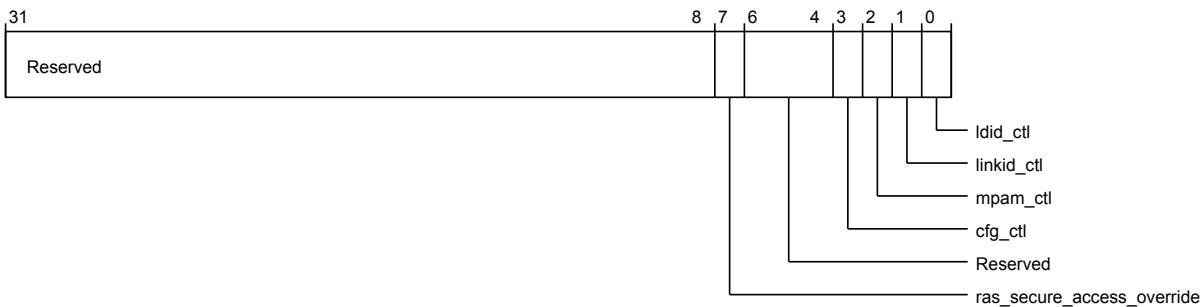


Figure 3-468 por_cxg_ha_por_cxg_ha_secure_register_groups_override (low)

The following table shows the por_cxg_ha_secure_register_groups_override lower register bit assignments.

Table 3-488 por_cxg_ha_por_cxg_ha_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0
6:4	Reserved	Reserved	RO	-
3	cfg_ctl	Allows non-secure access to secure HA config control registers	RW	1'b0
2	mpam_ctl	Allows non-secure access to secure HA MPAM override registers	RW	1'b0
1	linkid_ctl	Allows non-secure access to secure HA Link ID registers	RW	1'b0
0	ldid_ctl	Allows non-secure access to secure HA LDID registers	RW	1'b0

por_cxg_ha_unit_info

Provides component identification information for CXHA.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h900
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

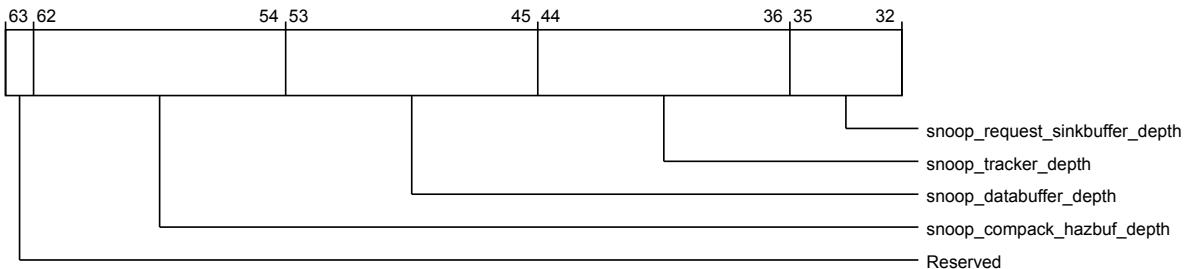


Figure 3-469 por_cxg_ha_por_cxg_ha_unit_info (high)

The following table shows the por_cxg_ha_unit_info higher register bit assignments.

Table 3-489 por_cxg_ha_por_cxg_ha_unit_info (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:54	snoop_compack_hazbuf_depth	Depth of CompAck snoop hazard buffer	RO	Configuration dependent
53:45	snoop_databuffer_depth	Depth of snoop data buffer	RO	Configuration dependent
44:36	snoop_tracker_depth	Depth of snoop tracker; number of outstanding SNP requests on CCIX	RO	Configuration dependent
35:32	snoop_request_sinkbuffer_depth	Depth of snoop request sink buffer; number of CHI SNP requests that can be sunk by CXHA	RO	Configuration dependent

The following image shows the lower register bit assignments.

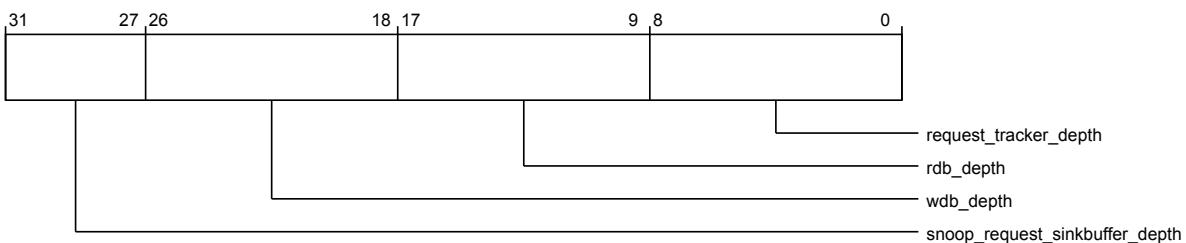


Figure 3-470 por_cxg_ha_por_cxg_ha_unit_info (low)

The following table shows the por cxg ha unit info lower register bit assignments.

Table 3-490 por_cxg_ha_por_cxg_ha_unit_info (low)

Bits	Field name	Description	Type	Reset
31:27	snoop_request_sinkbuffer_depth	Depth of snoop request sink buffer; number of CHI SNP requests that can be sunk by CXHA	RO	Configuration dependent
26:18	wdb_depth	Depth of write data buffer	RO	Configuration dependent
17:9	rdb_depth	Depth of read data buffer	RO	Configuration dependent
8:0	request_tracker_depth	Depth of request tracker	RO	Configuration dependent

por_cxg_ha_unit_info2

Provides additional component identification information for CXHA.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h908

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

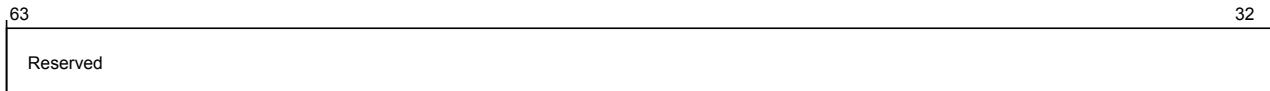


Figure 3-471 por_cxg_ha_por_cxg_ha_unit_info2 (high)

The following table shows the por_cxg_ha_unit_info2 higher register bit assignments.

Table 3-491 por_cxg_ha_por_cxg_ha_unit_info2 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

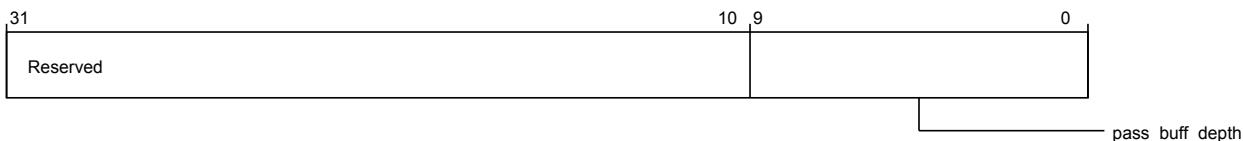


Figure 3-472 por_cxg_ha_por_cxg_ha_unit_info2 (low)

The following table shows the por_cxg_ha_unit_info2 lower register bit assignments.

Table 3-492 por_cxg_ha_por_cxg_ha_unit_info2 (low)

Bits	Field name	Description	Type	Reset
31:10	Reserved	Reserved	RO	-
9:0	pass_buff_depth	Depth of Passive Buffer	RO	Configuration dependent

por_cxg_ha_agentid_to_linkid_reg0

Specifies the mapping of Agent ID to Link ID for Agent IDs 0 to 7.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F00

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

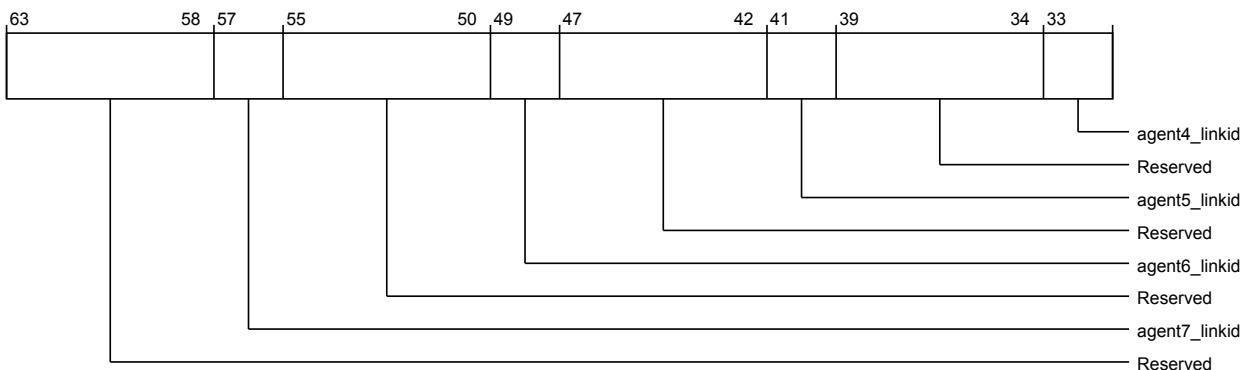


Figure 3-473 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg0 (high)

The following table shows the por_cxg_ha_agentid_to_linkid_reg0 higher register bit assignments.

Table 3-493 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg0 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent7_linkid	Specifies Link ID 7	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent6_linkid	Specifies Link ID 6	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent5_linkid	Specifies Link ID 5	RW	2'h0

Table 3-493 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg0 (high) (continued)

Bits	Field name	Description	Type	Reset
39:34	Reserved	Reserved	RO	-
33:32	agent4_linkid	Specifies Link ID 4	RW	2'h0

The following image shows the lower register bit assignments.

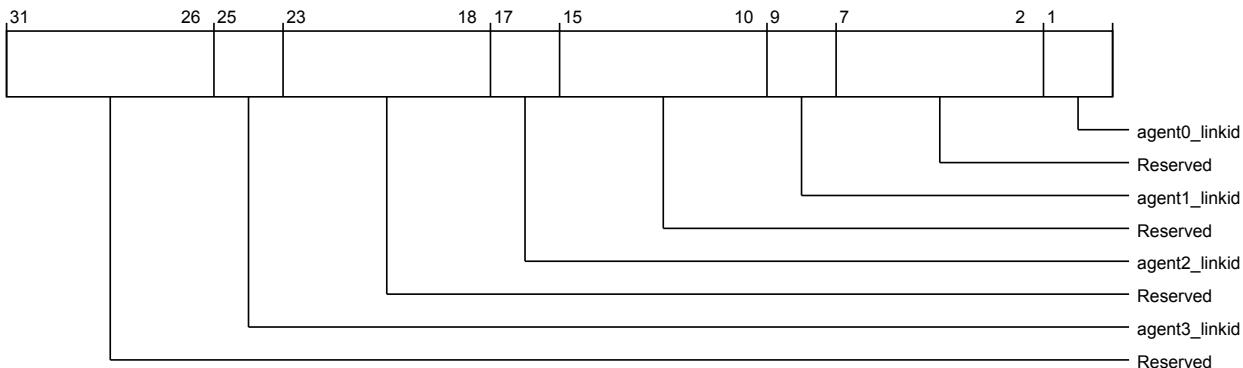


Figure 3-474 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg0 (low)

The following table shows the por_cxg_ha_agentid_to_linkid_reg0 lower register bit assignments.

Table 3-494 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg0 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent3_linkid	Specifies Link ID 3	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent2_linkid	Specifies Link ID 2	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent1_linkid	Specifies Link ID 1	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent0_linkid	Specifies Link ID 0	RW	2'h0

por_cxg_ha_agentid_to_linkid_reg1

Specifies the mapping of Agent ID to Link ID for Agent IDs 8 to 15.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1F08
Register reset	64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

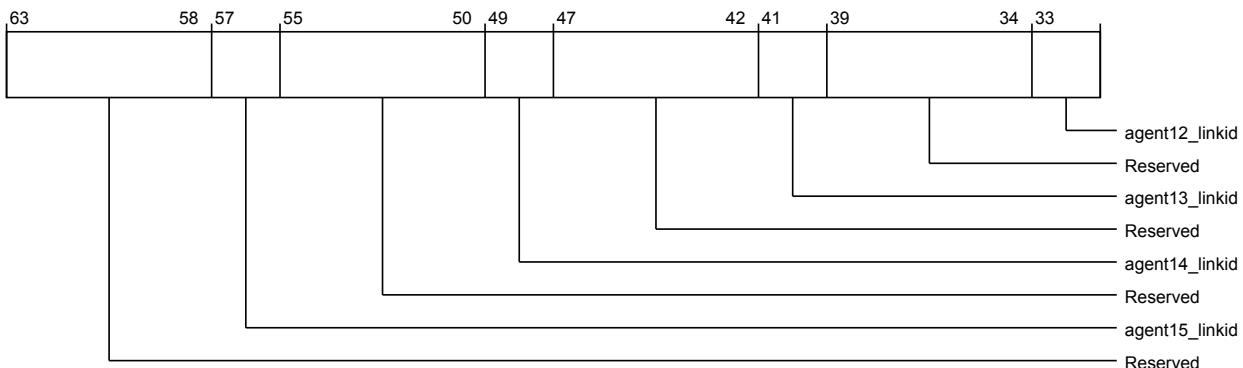


Figure 3-475 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg1 (high)

The following table shows the por_cxg_ha_agentid_to_linkid_reg1 higher register bit assignments.

Table 3-495 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg1 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent15_linkid	Specifies Link ID 15	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent14_linkid	Specifies Link ID 14	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent13_linkid	Specifies Link ID 13	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent12_linkid	Specifies Link ID 12	RW	2'h0

The following image shows the lower register bit assignments.

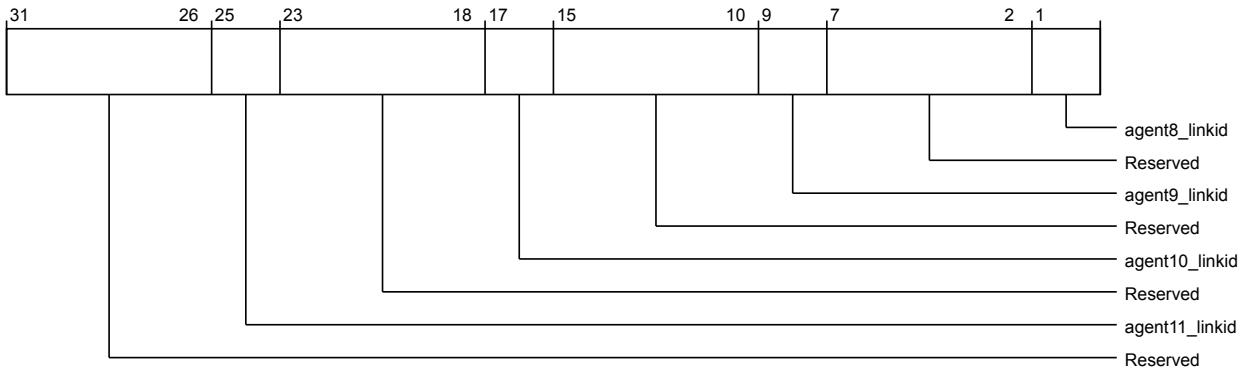


Figure 3-476 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg1 (low)

The following table shows the por_cxg_ha_agentid_to_linkid_reg1 lower register bit assignments.

Table 3-496 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg1 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent11_linkid	Specifies Link ID 11	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent10_linkid	Specifies Link ID 10	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent9_linkid	Specifies Link ID 9	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent8_linkid	Specifies Link ID 8	RW	2'h0

por_cxg_ha_agentid_to_linkid_reg2

Specifies the mapping of Agent ID to Link ID for Agent IDs 16 to 23.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F10

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

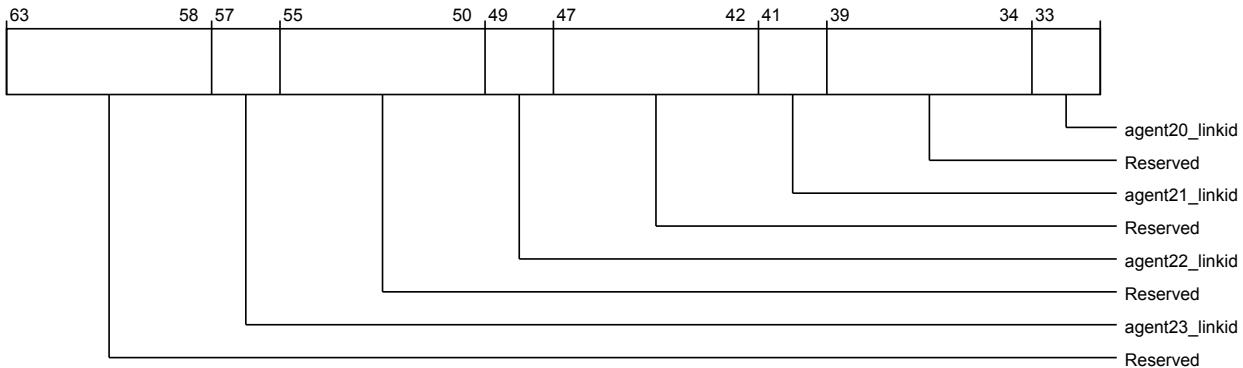


Figure 3-477 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg2 (high)

The following table shows the port assignments to higher register bit assignments.

Table 3-497 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg2 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent23_linkid	Specifies Link ID 23	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent22_linkid	Specifies Link ID 22	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent21_linkid	Specifies Link ID 21	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent20_linkid	Specifies Link ID 20	RW	2'h0

The following image shows the lower register bit assignments.

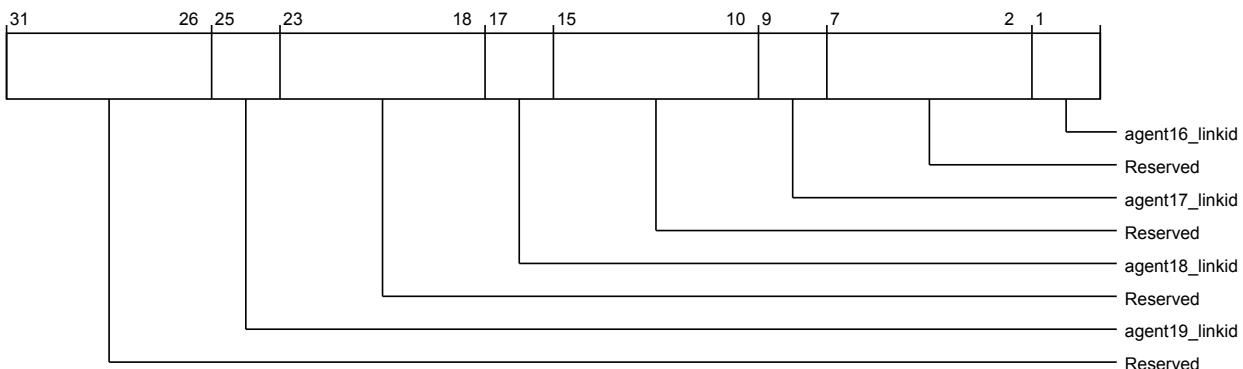


Figure 3-478 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg2 (low)

The following table shows the port cxg ha agentid to linkid reg2 lower register bit assignments.

Table 3-498 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg2 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent19_linkid	Specifies Link ID 19	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent18_linkid	Specifies Link ID 18	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent17_linkid	Specifies Link ID 17	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent16_linkid	Specifies Link ID 16	RW	2'h0

por_cxg_ha_agentid_to_linkid_reg3

Specifies the mapping of Agent ID to Link ID for Agent IDs 24 to 31.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F18

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

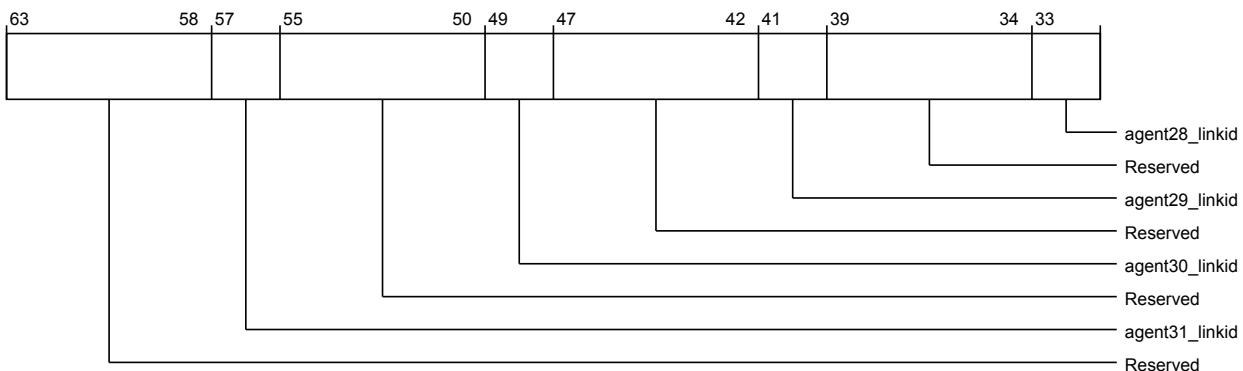


Figure 3-479 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg3 (high)

The following table shows the por_cxg_ha_agentid_to_linkid_reg3 higher register bit assignments.

Table 3-499 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg3 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent31_linkid	Specifies Link ID 31	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent30_linkid	Specifies Link ID 30	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent29_linkid	Specifies Link ID 29	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent28_linkid	Specifies Link ID 28	RW	2'h0

The following image shows the lower register bit assignments.

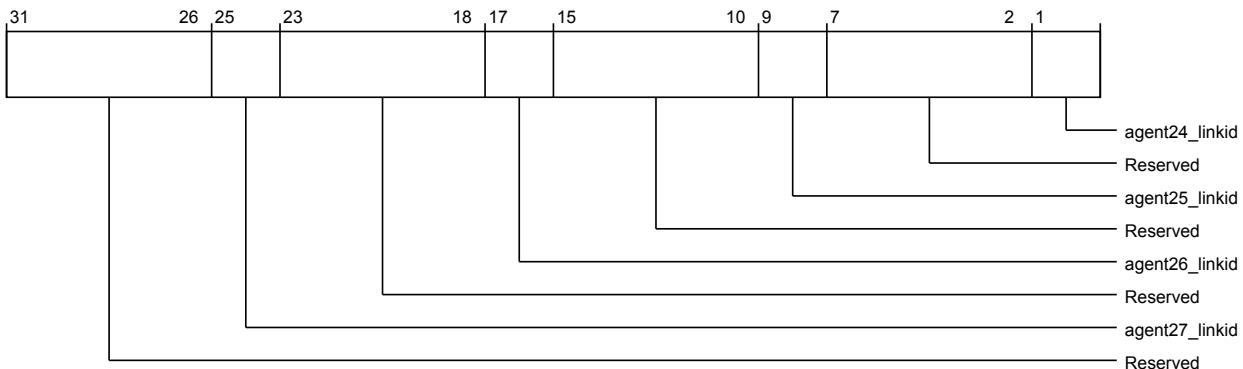


Figure 3-480 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg3 (low)

The following table shows the por_cxg_ha_agentid_to_linkid_reg3 lower register bit assignments.

Table 3-500 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg3 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent27_linkid	Specifies Link ID 27	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent26_linkid	Specifies Link ID 26	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent25_linkid	Specifies Link ID 25	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent24_linkid	Specifies Link ID 24	RW	2'h0

por_cxg_ha_agentid_to_linkid_reg4

Specifies the mapping of Agent ID to Link ID for Agent IDs 32 to 39.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1F20
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

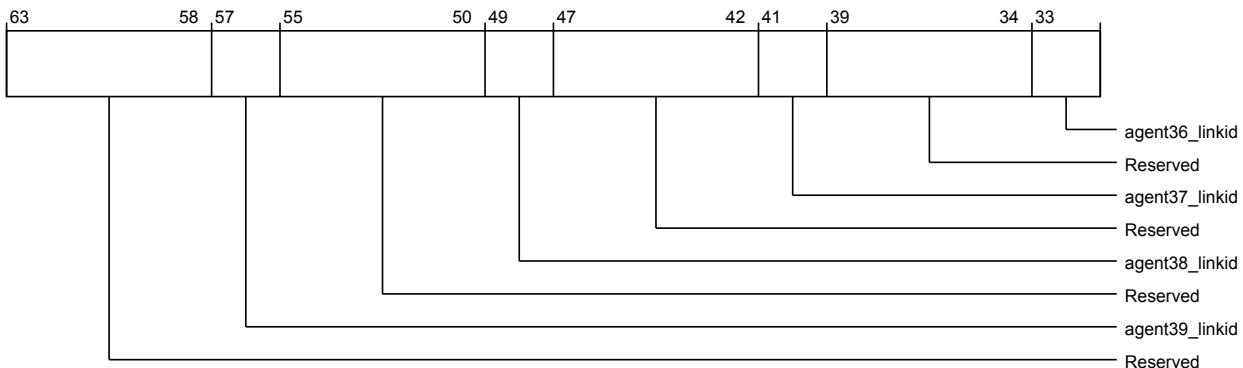


Figure 3-481 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg4 (high)

The following table shows the port cxg ha agentid to linkid reg4 higher register bit assignments.

Table 3-501 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg4 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent39_linkid	Specifies Link ID 39	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent38_linkid	Specifies Link ID 38	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent37_linkid	Specifies Link ID 37	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent36_linkid	Specifies Link ID 36	RW	2'h0

The following image shows the lower register bit assignments.

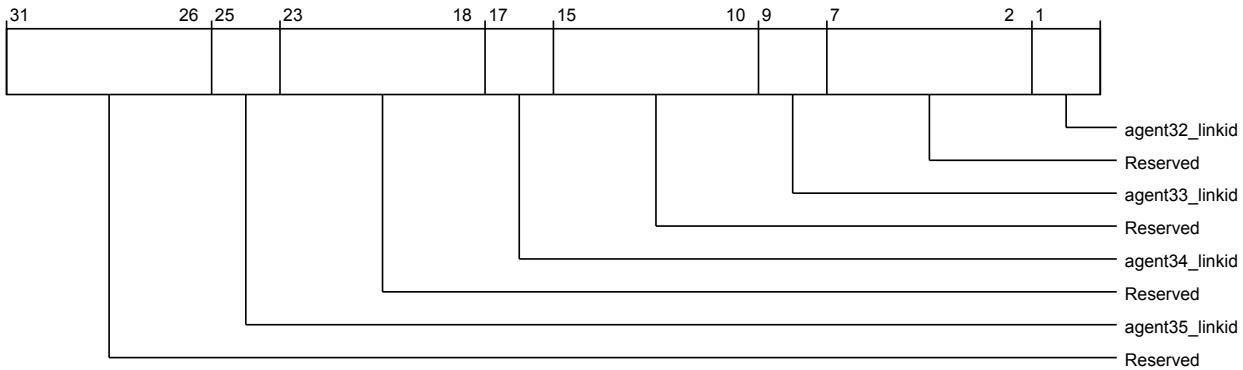


Figure 3-482 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg4 (low)

The following table shows the por_cxg_ha_agentid_to_linkid_reg4 lower register bit assignments.

Table 3-502 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg4 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent35_linkid	Specifies Link ID 35	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent34_linkid	Specifies Link ID 34	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent33_linkid	Specifies Link ID 33	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent32_linkid	Specifies Link ID 32	RW	2'h0

por_cxg_ha_agentid_to_linkid_reg5

Specifies the mapping of Agent ID to Link ID for Agent IDs 40 to 47.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F28

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

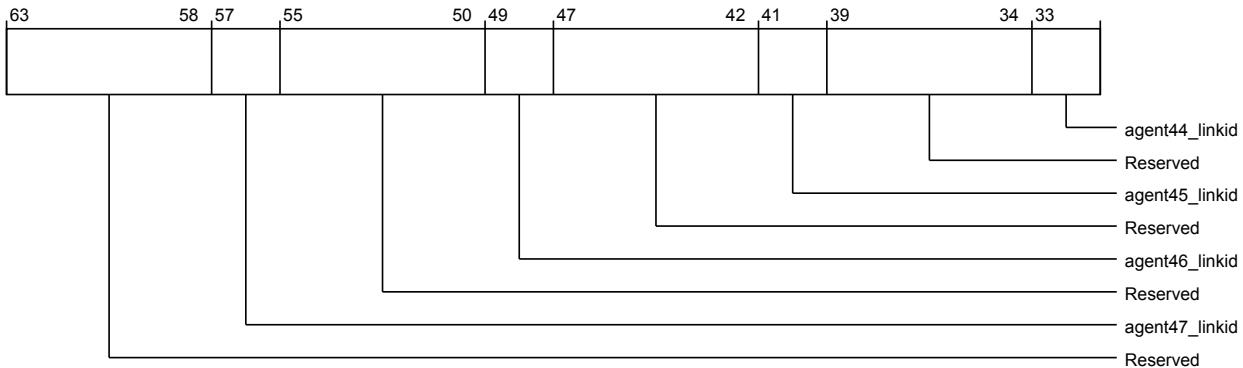


Figure 3-483 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg5 (high)

The following table shows the por_cxg_ha_agentid to linkid_reg5 higher register bit assignments.

Table 3-503 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg5 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent47_linkid	Specifies Link ID 47	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent46_linkid	Specifies Link ID 46	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent45_linkid	Specifies Link ID 45	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent44_linkid	Specifies Link ID 44	RW	2'h0

The following image shows the lower register bit assignments.

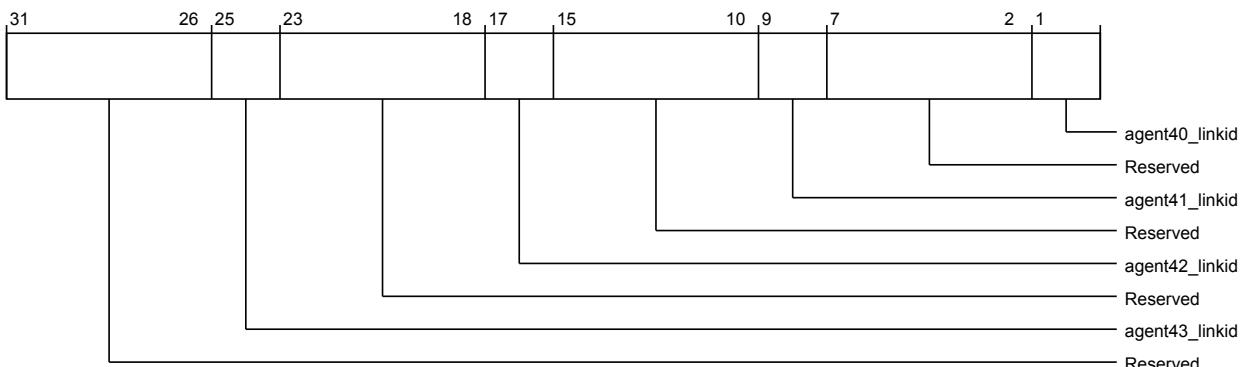


Figure 3-484 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg5 (low)

The following table shows the port cxg having agentid to linkid reg5 lower register bit assignments.

Table 3-504 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg5 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent43_linkid	Specifies Link ID 43	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent42_linkid	Specifies Link ID 42	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent41_linkid	Specifies Link ID 41	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent40_linkid	Specifies Link ID 40	RW	2'h0

por_cxg_ha_agentid_to_linkid_reg6

Specifies the mapping of Agent ID to Link ID for Agent IDs 48 to 55.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1F30

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

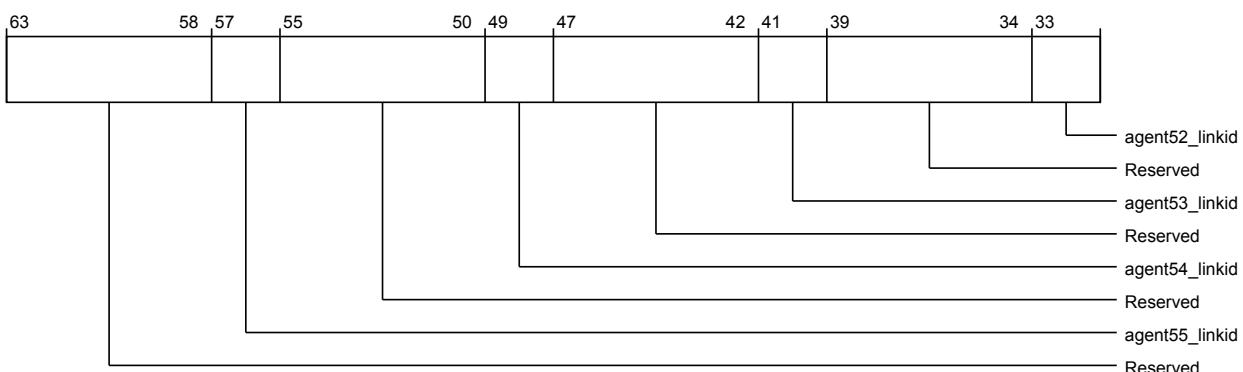


Figure 3-485 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg6 (high)

The following table shows the por_cxg_ha_agentid_to_linkid_reg6 higher register bit assignments.

Table 3-505 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg6 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent55_linkid	Specifies Link ID 55	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent54_linkid	Specifies Link ID 54	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent53_linkid	Specifies Link ID 53	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent52_linkid	Specifies Link ID 52	RW	2'h0

The following image shows the lower register bit assignments.

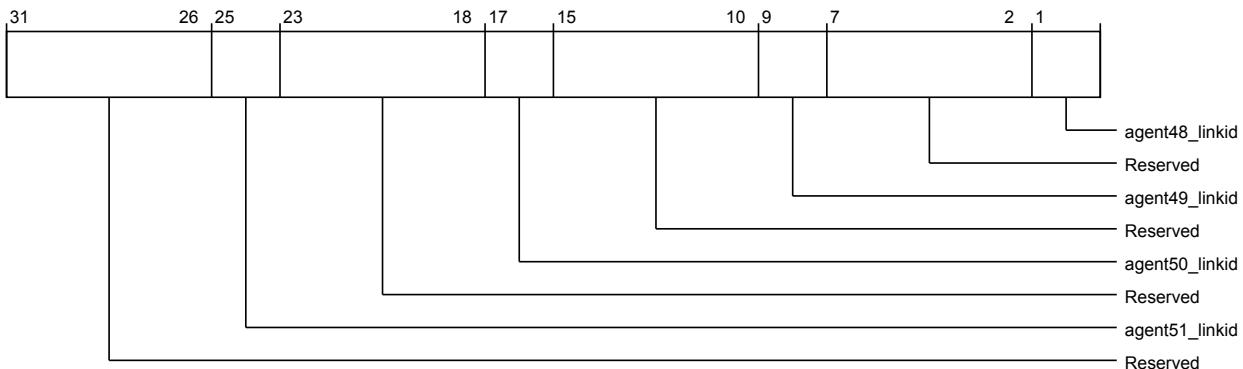


Figure 3-486 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg6 (low)

The following table shows the por_cxg_ha_agentid_to_linkid_reg6 lower register bit assignments.

Table 3-506 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg6 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent51_linkid	Specifies Link ID 51	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent50_linkid	Specifies Link ID 50	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent49_linkid	Specifies Link ID 49	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent48_linkid	Specifies Link ID 48	RW	2'h0

por_cxg_ha_agentid_to_linkid_reg7

Specifies the mapping of Agent ID to Link ID for Agent IDs 56 to 63.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1F38
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

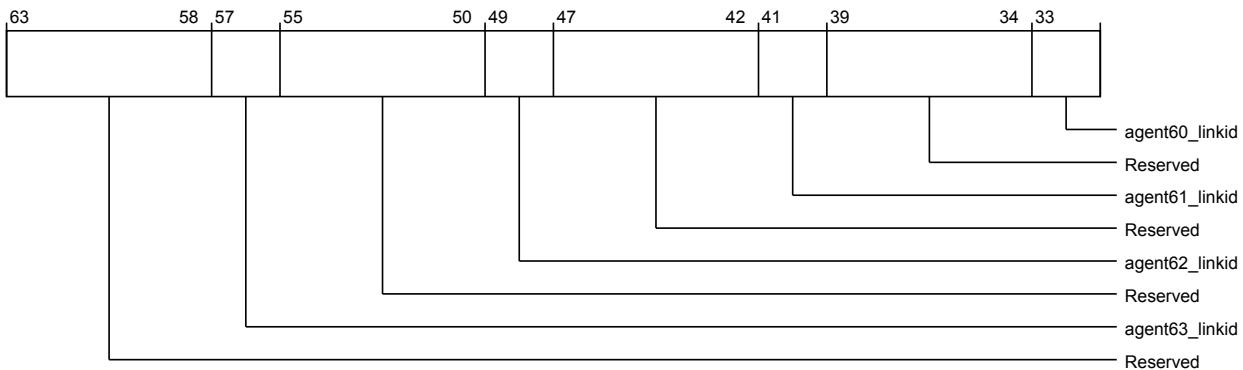


Figure 3-487 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg7 (high)

The following table shows the por_cxg_ha_agentid_to_linkid_reg7 higher register bit assignments.

Table 3-507 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg7 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent63_linkid	Specifies Link ID 63	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent62_linkid	Specifies Link ID 62	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent61_linkid	Specifies Link ID 61	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent60_linkid	Specifies Link ID 60	RW	2'h0

The following image shows the lower register bit assignments.

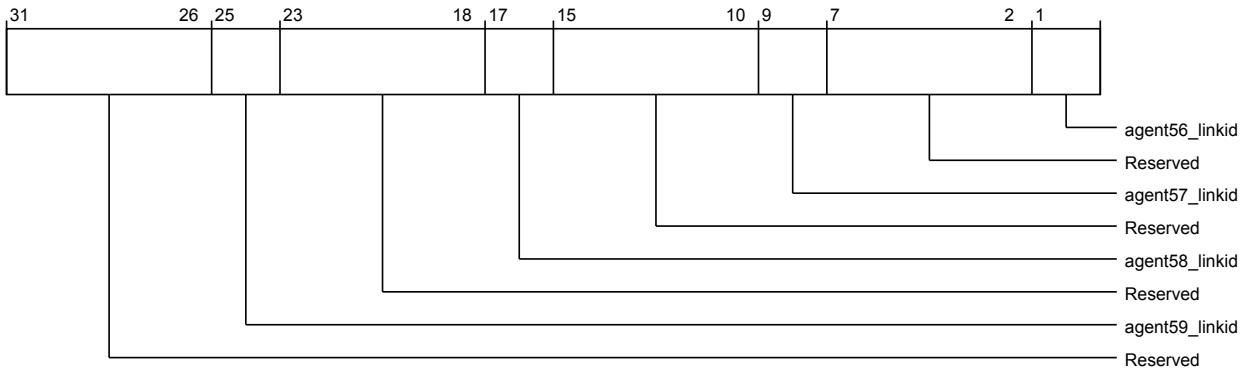


Figure 3-488 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg7 (low)

The following table shows the por_cxg_ha_agentid_to_linkid_reg7 lower register bit assignments.

Table 3-508 por_cxg_ha_por_cxg_ha_agentid_to_linkid_reg7 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent59_linkid	Specifies Link ID 59	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent58_linkid	Specifies Link ID 58	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent57_linkid	Specifies Link ID 57	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent56_linkid	Specifies Link ID 56	RW	2'h0

por_cxg_ha_agentid_to_linkid_val

Specifies which Agent ID to Link ID mappings are valid.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1FF8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

63
valid

32

Figure 3-489 por_cxg_ha_por_cxg_ha_agentid_to_linkid_val (high)

The following table shows the por_cxg_ha_agentid_to_linkid_val higher register bit assignments.

Table 3-509 por_cxg_ha_por_cxg_ha_agentid_to_linkid_val (high)

Bits	Field name	Description	Type	Reset
63:32	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

The following image shows the lower register bit assignments.

31
valid

0

Figure 3-490 por_cxg_ha_por_cxg_ha_agentid_to_linkid_val (low)

The following table shows the por_cxg_ha_agentid_to_linkid_val lower register bit assignments.

Table 3-510 por_cxg_ha_por_cxg_ha_agentid_to_linkid_val (low)

Bits	Field name	Description	Type	Reset
31:0	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255

This register repeats 255 times. It parametrized by the index from 0 to 255. Specifies the mapping of Expanded RAID to RN-F LDID for Expanded RAIDs #{{index}*4} to #{{index}*4+3}.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC00 + #{{8*[0, 1, 2, ..., 254, 255]}}

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cxg_ha_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

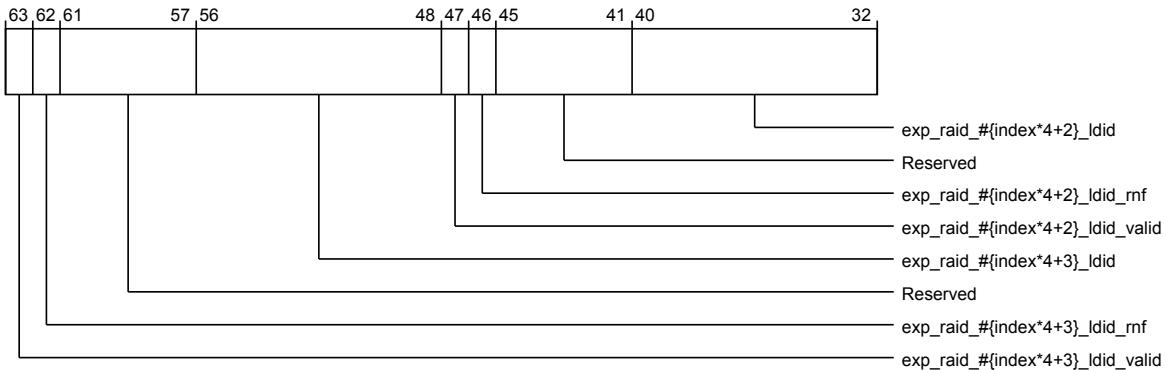


Figure 3-491 por_cxg_ha_por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255 (high)

The following table shows the por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255 higher register bit assignments.

Table 3-511 por_cxg_ha_por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255 (high)

Bits	Field name	Description	Type	Reset
63	exp_raid_{index*4+3}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4+3} is valid	RW	1'b0
62	exp_raid_{index*4+3}_ldid_mf	Specifies if Expanded RAID #{index*4+3} is RN-F	RW	1'b0
61:57	Reserved	Reserved	RO	-
56:48	exp_raid_{index*4+3}_ldid	Specifies the LDID for Expanded RAID #{index*4+3}	RW	9'h0
47	exp_raid_{index*4+2}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4+2} is valid	RW	1'b0
46	exp_raid_{index*4+2}_ldid_mf	Specifies if Expanded RAID #{index*4+2} is RN-F	RW	1'b0
45:41	Reserved	Reserved	RO	-
40:32	exp_raid_{index*4+2}_ldid	Specifies the LDID for Expanded RAID #{index*4+2}	RW	9'h0

The following image shows the lower register bit assignments.

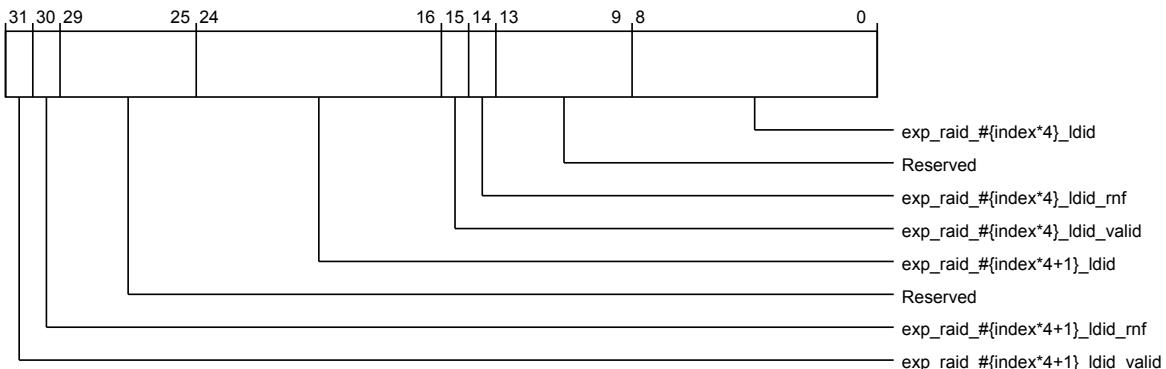


Figure 3-492 por_cxg_ha_por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255 (low)

The following table shows the por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255 lower register bit assignments.

Table 3-512 por_cxg_ha_por_cxg_ha_rnf_exp_raid_to_ldid_reg_0-255 (low)

Bits	Field name	Description	Type	Reset
31	exp_raid_{index*4+1}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4+1} is valid	RW	1'b0
30	exp_raid_{index*4+1}_ldid_rnf	Specifies if Expanded RAID #{index*4+1} is RN-F	RW	1'b0
29:25	Reserved	Reserved	RO	-
24:16	exp_raid_{index*4+1}_ldid	Specifies the LDID for Expanded RAID #{index*4+1}	RW	9'h0
15	exp_raid_{index*4}_ldid_valid	Specifies if LDID for Expanded RAID #{index*4} is valid	RW	1'b0
14	exp_raid_{index*4}_ldid_rnf	Specifies if Expanded RAID #{index*4} is RN-F	RW	1'b0
13:9	Reserved	Reserved	RO	-
8:0	exp_raid_{index*4}_ldid	Specifies the LDID for Expanded RAID #{index*4}	RW	9'h0

por_cxg_ha_pmu_event_sel

Specifies the PMU event to be counted as a 6-bit ID with the following encodings: 6'b000000:
 CXHA_PMU_EVENT_NULL 6'b100001: CXHA_PMU_EVENT_RDDATBYP 6'b100010:
 CXHA_PMU_EVENT_CHIRSP_UP_STALL 6'b100011:
 CXHA_PMU_EVENT_CHIDAT_UP_STALL 6'b100100:
 CXHA_PMU_EVENT_SNPPCRD_LNK0_STALL 6'b100101:
 CXHA_PMU_EVENT_SNPPCRD_LNK1_STALL 6'b100110:
 CXHA_PMU_EVENT_SNPPCRD_LNK2_STALL 6'b100111: CXHA_PMU_EVENT_REQTRK_OCC
 6'b101000: CXHA_PMU_EVENT_RDB_OCC 6'b101001: CXHA_PMU_EVENT_RDBBYP_OCC
 6'b101010: CXHA_PMU_EVENT_WDB_OCC 6'b101011: CXHA_PMU_EVENT_SNPTRK_OCC
 6'b101100: CXHA_PMU_EVENT_SDB_OCC 6'b101101: CXHA_PMU_EVENT_SNPHAZ_OCC

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2000
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

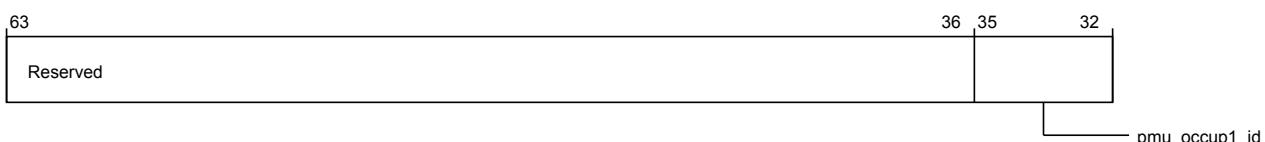


Figure 3-493 por_cxg_ha_por_cxg_ha_pmu_event_sel (high)

The following table shows the por_cxg_ha_pmu_event_sel higher register bit assignments.

Table 3-513 por_cxg_ha_por_cxg_ha_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:36	Reserved	Reserved	RO	-
35:32	pmu_occup1_id	CXHA PMU occupancy event selector ID	RW	4'b0

The following image shows the lower register bit assignments.

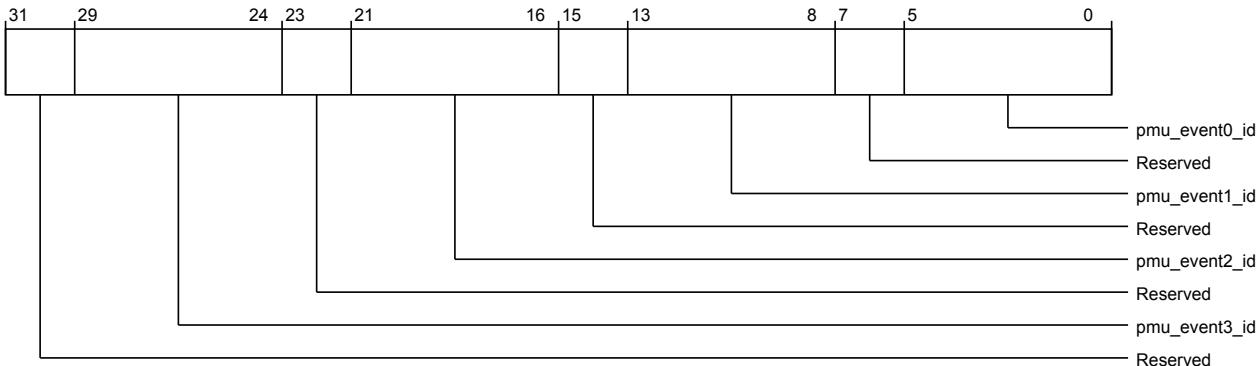


Figure 3-494 por_cxg_ha_por_cxg_ha_pmu_event_sel (low)

The following table shows the por_cxg_ha_pmu_event_sel lower register bit assignments.

Table 3-514 por_cxg_ha_por_cxg_ha_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	CXHA PMU Event 3 ID	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	CXHA PMU Event 2 ID	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	CXHA PMU Event 1 ID	RW	6'b0
7:6	Reserved	Reserved	RO	-
5:0	pmu_event0_id	CXHA PMU Event 0 ID	RW	6'b0

por_cxg_ha_cxprtcl_link0_ctl

Functions as the CXHA CCIX Protocol Link 0 control register. Works with por_cxg_ha_cxprtcl_link0_status.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1C00

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

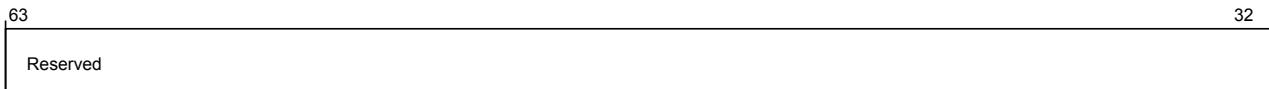


Figure 3-495 por_cxg_ha_por_cxg_ha_cxprtcl_link0_ctl (high)

The following table shows the por_cxg_ha_cxprtcl_link0_ctl higher register bit assignments.

Table 3-515 por_cxg_ha_por_cxg_ha_cxprtcl_link0_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

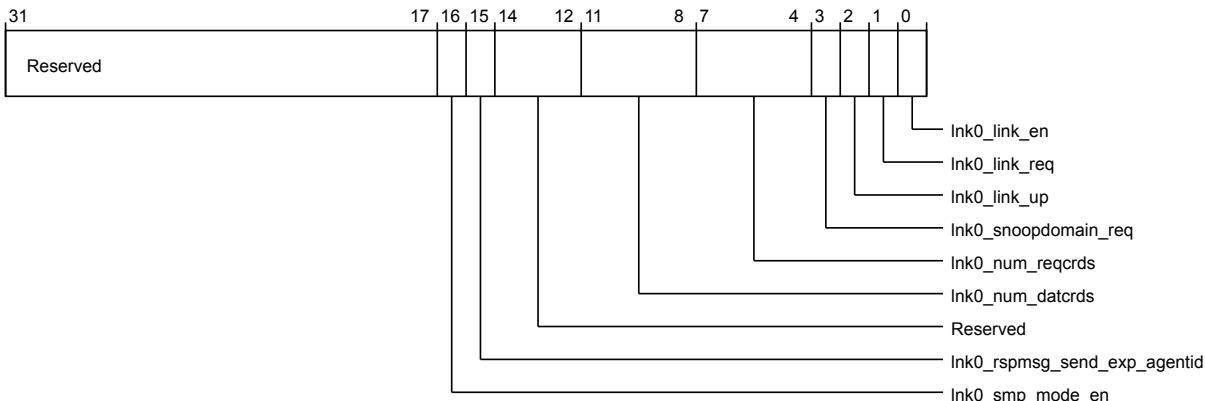


Figure 3-496 por_cxg_ha_por_cxg_ha_cxprtcl_link0_ctl (low)

The following table shows the por_cxg_ha_cxprtcl_link0_ctl lower register bit assignments.

Table 3-516 por_cxg_ha_por_cxg_ha_cxprtcl_link0_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	Inlk0_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 0.	RW	Configuration dependent
15	Inlk0_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 0	RW	1'b0
14:12	Reserved	Reserved	RO	-

Table 3-516 por_cxg_ha_por_cxg_ha_cxprtl_link0_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
11:8	lnk0_num_daterds	Controls the number of CCIX data credits assigned to Link 0 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
7:4	lnk0_num_reqcrds	Controls the number of CCIX request credits assigned to Link 0 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
3	lnk0_snoopdomain_req	Controls Snoop domain enable (SYSCOREQ) for CCIX Link 0	RW	1'b0
2	lnk0_link_up	Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent 1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear 1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent	RW	1'b0
1	lnk0_link_req	Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent 1'b0: Link Down request NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state. 1'b1: Link Up request	RW	1'b0
0	lnk0_link_en	Enables CCIX Link 0 when set 1'b0: Link is disabled 1'b1: Link is enabled	RW	1'b0

por_cxg_ha_cxpctl_link0_status

Functions as the CXHA CCIX Protocol Link 0 status register. Works with por_cxg_ha_cxpctl_link0_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C08

Register reset 64'b010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-497 por_cxg_ha_por_cxg_ha_cxpctl_link0_status (high)

The following table shows the por_cxg_ha_cxpctl_link0_status higher register bit assignments.

Table 3-517 por_cxg_ha_por_cxg_ha_cxpctl_link0_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

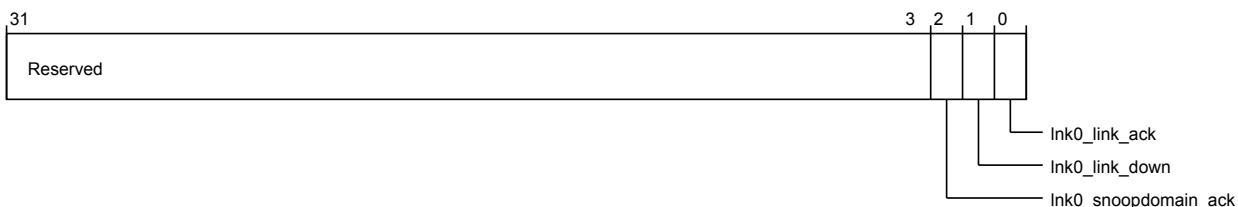


Figure 3-498 por_cxg_ha_por_cxg_ha_cxpctl_link0_status (low)

The following table shows the por_cxg_ha_cxpctl_link0_status lower register bit assignments.

Table 3-518 por_cxg_ha_por_cxg_ha_cxpctl_link0_status (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	Lnk0_snoopdomain_ack	Provides Snoop domain status (SYSCOACK) for CCIX Link 0	RO	1'b0

Table 3-518 por_cxg_ha_por_cxg_ha_cxpctl_link0_status (low) (continued)

Bits	Field name	Description	Type	Reset
1	lnk0_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	lnk0_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_cxg_ha_cxpctl_link1_ctl

Functions as the CXHA CCIX Protocol Link 1 control register. Works with por_cxg_ha_cxpctl_link1_status.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C10
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-499 por_cxg_ha_por_cxg_ha_cxpctl_link1_ctl (high)

The following table shows the por_cxg_ha_cxpctl_link1_ctl higher register bit assignments.

Table 3-519 por_cxg_ha_por_cxg_ha_cxpctl_link1_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

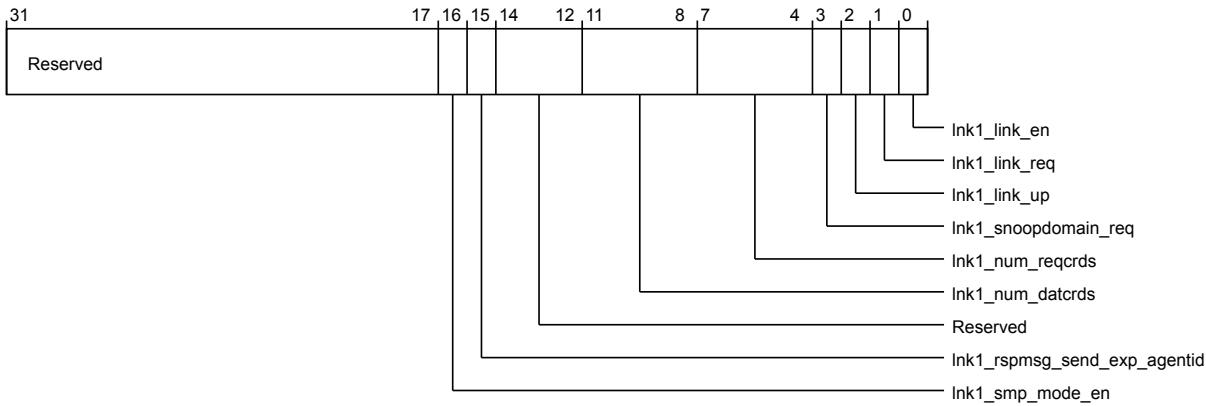


Figure 3-500 por_cxg_ha_por_cxg_ha_cxprtcl_link1_ctl (low)

The following table shows the por_cxg_ha_cxprtcl_link1_ctl lower register bit assignments.

Table 3-520 por_cxg_ha_por_cxg_ha_cxprtcl_link1_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	lnk1_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 1.	RW	Configuration dependent
15	lnk1_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 1	RW	1'b0
14:12	Reserved	Reserved	RO	-
11:8	lnk1_num_datcrds	Controls the number of CCIX data credits assigned to Link 1 4'h0: Total credits equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
7:4	lnk1_num_reqcrds	Controls the number of CCIX request credits assigned to Link 1 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
3	lnk1_snoopdomain_req	Controls Snoop domain enable (SYSCOREQ) for CCIX Link 1	RW	1'b0

Table 3-520 por_cxg_ha_por_cxg_ha_cxprtcl_link1_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	lnk1_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk1_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk1_link_en	<p>Enables CCIX Link 1 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_cxg_ha_cxprtcl_link1_status

Functions as the CXHA CCIX Protocol Link 1 status register. Works with por_cxg_ha_cxprtcl_link1_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C18

Register reset 64'b010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-501 por_cxg_ha_por_cxg_ha_cxprtcl_link1_status (high)

The following table shows the por_cxg_ha_cxprtcl_link1_status higher register bit assignments.

Table 3-521 por_cxg_ha_por_cxg_ha_cxprtcl_link1_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

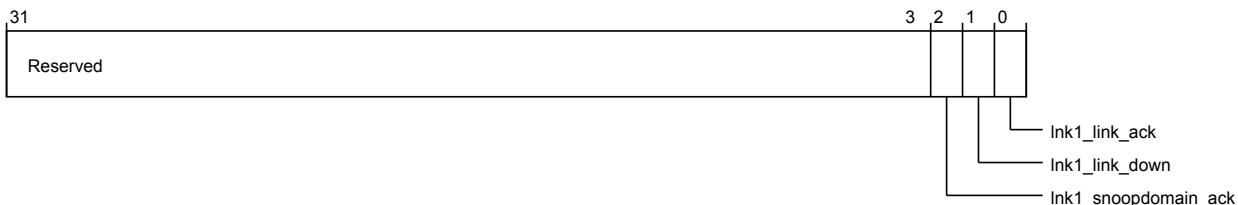


Figure 3-502 por_cxg_ha_por_cxg_ha_cxprtcl_link1_status (low)

The following table shows the por_cxg_ha_cxprtcl_link1_status lower register bit assignments.

Table 3-522 por_cxg_ha_por_cxg_ha_cxprtcl_link1_status (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	Lnk1_snoopdomain_ack	Provides Snoop domain status (SYSCOACK) for CCIX Link 1	RO	1'b0
1	Lnk1_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	Lnk1_link_ack	Link Up/Down Acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_cxg_ha_cxprtcl_link2_ctl

Functions as the CXHA CCIX Protocol Link 2 control register. Works with por_cxg_ha_cxprtcl_link2_status.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C20

Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

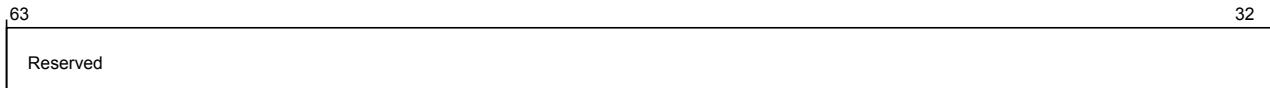


Figure 3-503 por_cxg_ha_por_cxg_ha_cxpctl_link2_ctl (high)

The following table shows the por_cxg_ha_cxpctl_link2_ctl higher register bit assignments.

Table 3-523 por_cxg_ha_por_cxg_ha_cxpctl_link2_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

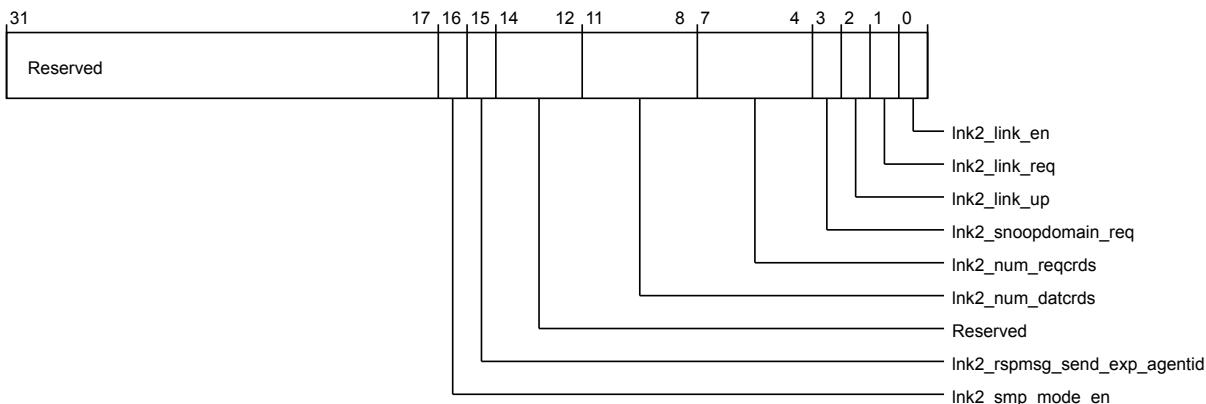


Figure 3-504 por_cxg_ha_por_cxg_ha_cxpctl_link2_ctl (low)

The following table shows the por_cxg_ha_cxpctl_link2_ctl lower register bit assignments.

Table 3-524 por_cxg_ha_por_cxg_ha_cxpctl_link2_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	Link2_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 2.	RW	Configuration dependent
15	Link2_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 2	RW	1'b0
14:12	Reserved	Reserved	RO	-

Table 3-524 por_cxg_ha_por_cxg_ha_cxprtcl_link2_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
11:8	lnk2_num_daterds	Controls the number of CCIX data credits assigned to Link 2 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
7:4	lnk2_num_reqcrds	Controls the number of CCIX request credits assigned to Link 2 4'h0: Total credits are equally divided across all links 4'h1: 25% of credits assigned 4'h2: 50% of credits assigned 4'h3: 75% of credits assigned 4'h4: 100% of credits assigned 4'hF: 0% of credits assigned	RW	4'b0
3	lnk2_snoopdomain_req	Controls Snoop domain enable (SYSCOREQ) for CCIX Link 2	RW	1'b0
2	lnk2_link_up	Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent 1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear 1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent	RW	1'b0
1	lnk2_link_req	Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent 1'b0: Link Down request NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state. 1'b1: Link Up request	RW	1'b0
0	lnk2_link_en	Enables CCIX Link 2 when set 1'b0: Link is disabled 1'b1: Link is enabled	RW	1'b0

por_cxg_ha_cxprtcl_link2_status

Functions as the CXHA CCIX Protocol Link 2 status register. Works with por_cxg_ha_cxprtcl_link2_ctl.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h1C28
Register reset	64'b010
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

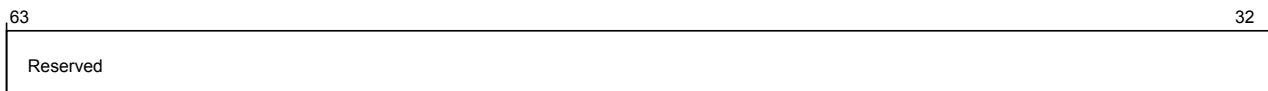


Figure 3-505 por_cxg_ha_por_cxg_ha_cxpctl_link2_status (high)

The following table shows the por_cxg_ha_cxpctl_link2_status higher register bit assignments.

Table 3-525 por_cxg_ha_por_cxg_ha_cxpctl_link2_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

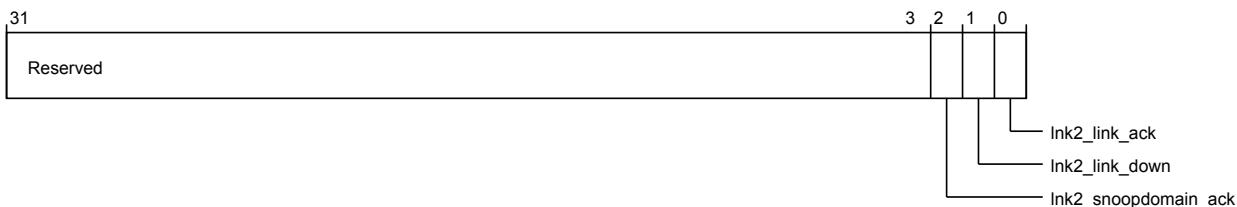


Figure 3-506 por_cxg_ha_por_cxg_ha_cxpctl_link2_status (low)

The following table shows the por_cxg_ha_cxpctl_link2_status lower register bit assignments.

Table 3-526 por_cxg_ha_por_cxg_ha_cxpctl_link2_status (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	lnk2_snoopdomain_ack	Provides Snoop domain status (SYSCOACK) for CCIX Link 2	RO	1'b0

Table 3-526 por_cxg_ha_por_cxg_ha_cxprtcl_link2_status (low) (continued)

Bits	Field name	Description	Type	Reset
1	lnk2_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	lnk2_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_cxg_ha_errfr

Functions as the error feature register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3000
Register reset	64'b0000010100101
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

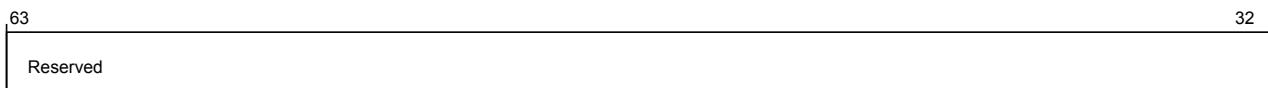


Figure 3-507 por_cxg_ha_por_cxg_ha_errfr (high)

The following table shows the por_cxg_ha_errfr higher register bit assignments.

Table 3-527 por_cxg_ha_por_cxg_ha_errfr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

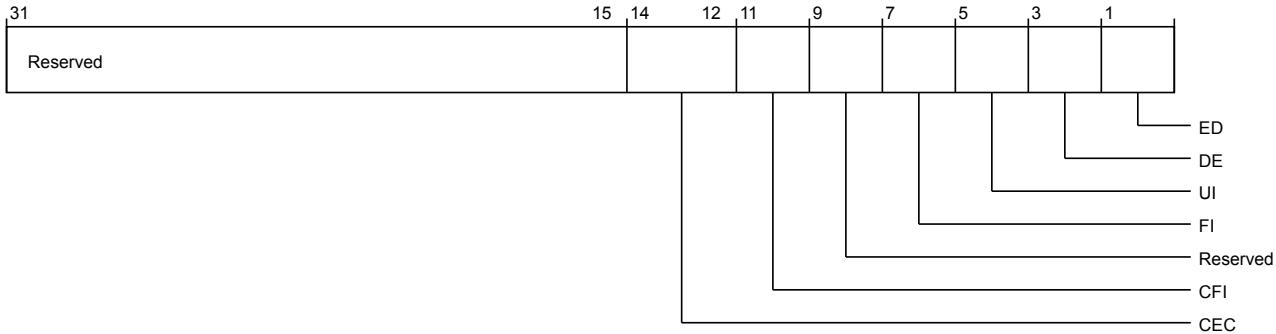


Figure 3-508 por_cxg_ha_por_cxg_ha_errfr (low)

The following table shows the por_cxg_ha_errfr lower register bit assignments.

Table 3-528 por_cxg_ha_por_cxg_ha_errfr (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_cxg_ha_errmisc[39:32] 3'b100: Implements 16-bit error counter in por_cxg_ha_errmisc[47:32]	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_cxg_ha_errctlr

Functions as the error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3008
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

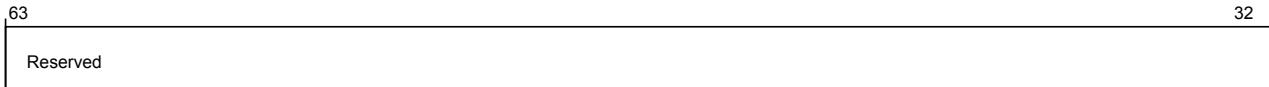


Figure 3-509 por_cxg_ha_por_cxg_ha_errctlr (high)

The following table shows the por_cxg_ha_errctlr higher register bit assignments.

Table 3-529 por_cxg_ha_por_cxg_ha_errctlr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

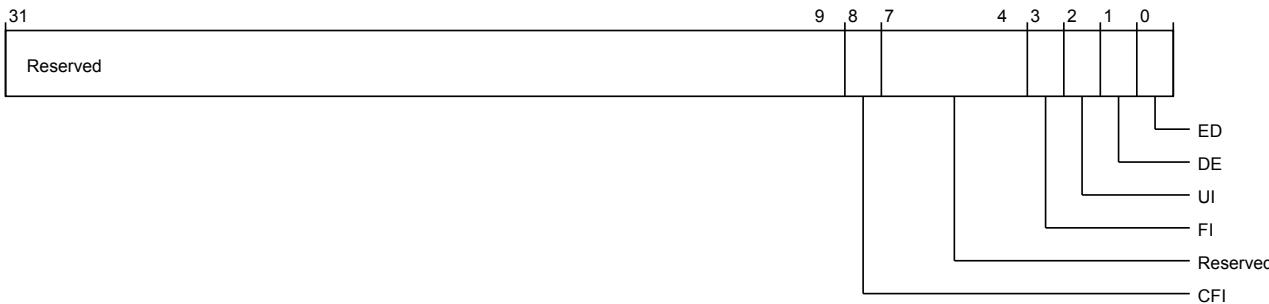


Figure 3-510 por_cxg_ha_por_cxg_ha_errctlr (low)

The following table shows the por_cxg_ha_errctlr lower register bit assignments.

Table 3-530 por_cxg_ha_por_cxg_ha_errctlr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_cxg_ha_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_cxg_ha_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_cxg_ha_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_cxg_ha_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_cxg_ha_errfr.ED	RW	1'b0

por_cxg_ha_errstatus

Functions as the error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3010
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

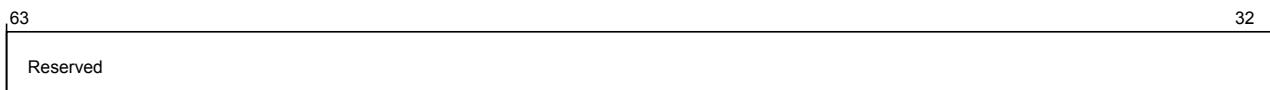


Figure 3-511 por_cxg_ha_por_cxg_ha_errstatus (high)

The following table shows the por_cxg_ha_errstatus higher register bit assignments.

Table 3-531 por_cxg_ha_por_cxg_ha_errstatus (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

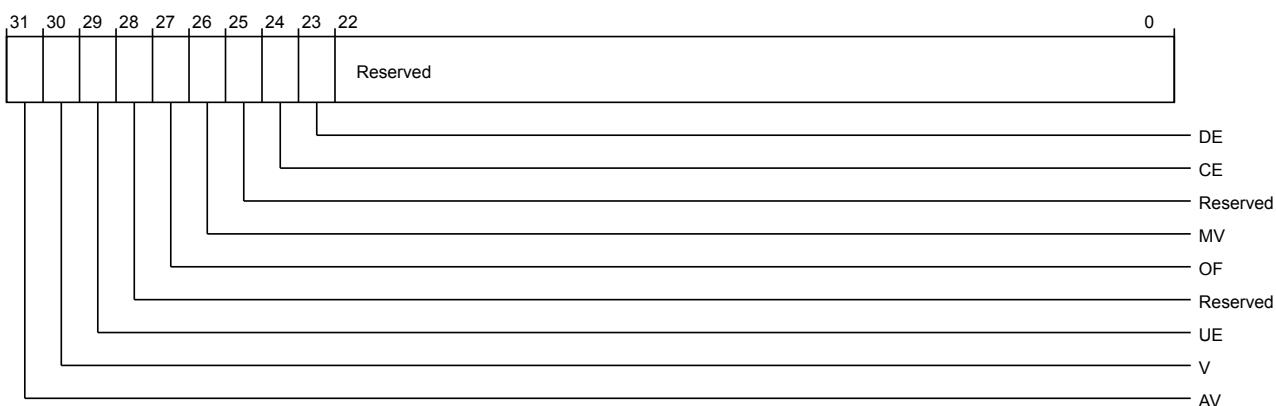


Figure 3-512 por_cxg_ha_por_cxg_ha_errstatus (low)

The following table shows the por_cxg_ha_errstatus lower register bit assignments.

Table 3-532 por_cxg_ha_por_cxg_ha_errstatus (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_cxg_ha_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_cxg_ha_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_cxg_ha_erraddr

Contains the error record address.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3018
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cxg_ha_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

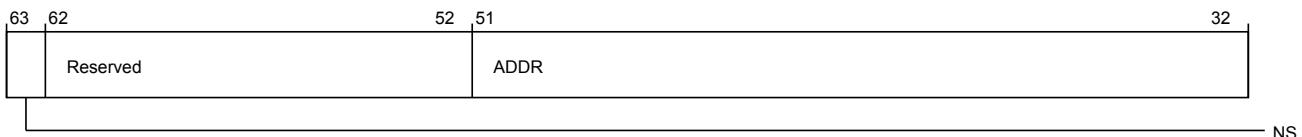


Figure 3-513 por_cxg_ha_por_cxg_ha_erraddr (high)

The following table shows the por_cxg_ha_erraddr higher register bit assignments.

Table 3-533 por_cxg_ha_por_cxg_ha_erraddr (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_cxg_ha_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.



Figure 3-514 por_cxg_ha_por_cxg_ha_erraddr (low)

The following table shows the por_cxg_ha_erraddr lower register bit assignments.

Table 3-534 por_cxg_ha_por_cxg_ha_erraddr (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_cxg_ha_errmisc

Functions as the miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3020
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group	por_cxg_ha_secure_register_groups_override.ras_secure_access_override_override

The following image shows the higher register bit assignments.

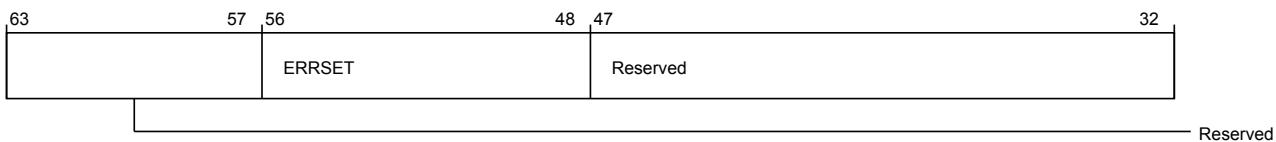


Figure 3-515 por_cxg_ha_por_cxg_ha_errmisc (high)

The following table shows the por_cxg_ha_errmisc higher register bit assignments.

Table 3-535 por_cxg_ha_por_cxg_ha_errmisc (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:48	ERRSET	RAM entry set address for parity error	RW	9'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

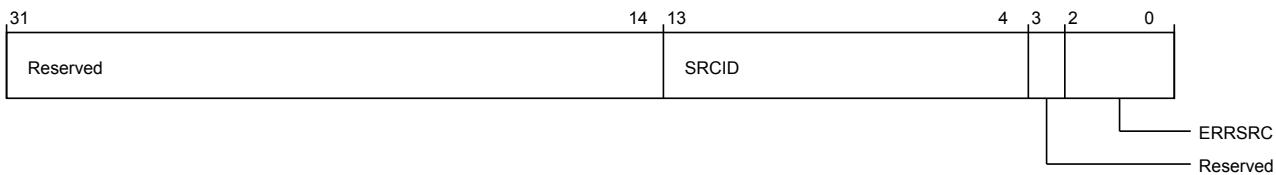


Figure 3-516 por_cxg_ha_por_cxg_ha_errmisc (low)

The following table shows the por_cxg_ha_errmisc lower register bit assignments.

Table 3-536 por_cxg_ha_por_cxg_ha_errmisc (low)

Bits	Field name	Description	Type	Reset
31:14	Reserved	Reserved	RO	-
13:4	SRCID	CCIX RAID of the requestor or the snoop target	RW	10'b0
3	Reserved	Reserved	RO	-
2:0	ERRSRC	Source of the parity error 3'b000: Read data buffer 0 3'b001: Read data buffer 1 3'b010: Write data buffer 0 3'b011: Write data buffer 1 3'b100: Passive Buffer	RW	3'b000

por_cxg_ha_errfr_NS

Functions as the non-secure error feature register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3100

Register reset 64'b0000010100101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

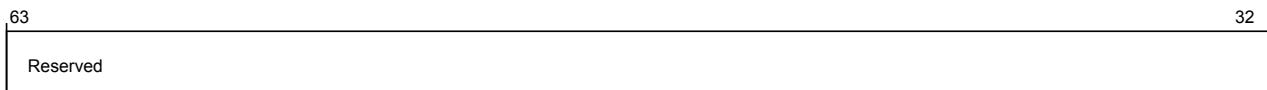


Figure 3-517 por_cxg_ha_por_cxg_ha_errfr_ns (high)

The following table shows the por_cxg_ha_errfr_NS higher register bit assignments.

Table 3-537 por_cxg_ha_por_cxg_ha_errfr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

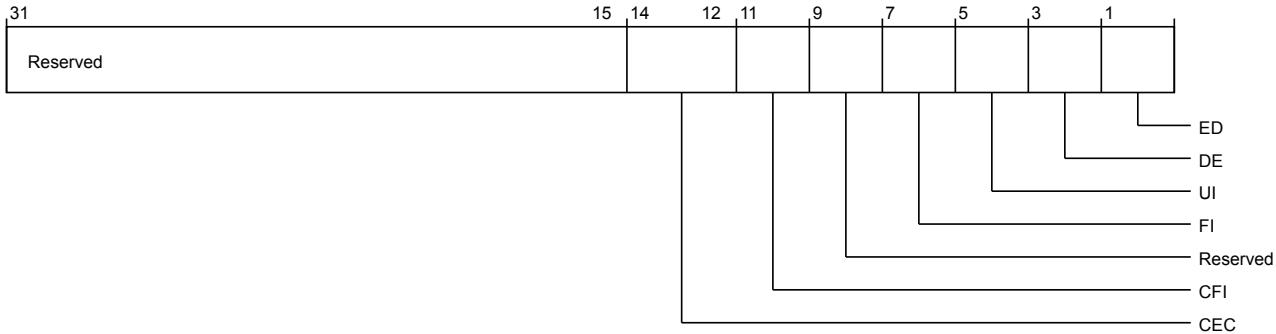


Figure 3-518 por_cxg_ha_por_cxg_ha_errfr_ns (low)

The following table shows the por_cxg_ha_errfr_NS lower register bit assignments.

Table 3-538 por_cxg_ha_por_cxg_ha_errfr_ns (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_cxg_ha_errmisc[39:32] 3'b100: Implements 16-bit error counter in por_cxg_ha_errmisc[47:32]	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_cxg_ha_errctlr_NS

Functions as the non-secure error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3

Register reset 64'b0

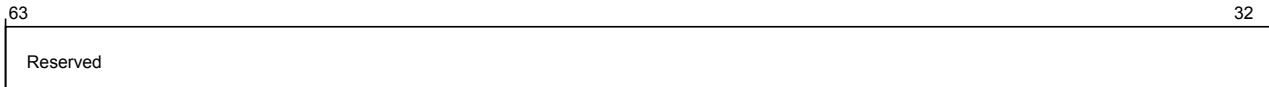


Figure 3-519 por_cxg_ha_por_cxg_ha_errctlr_ns (high)

The following table shows the por_cxg_ha_errctlr_NS higher register bit assignments.

Table 3-539 por_cxg_ha_por_cxg_ha_errctlr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

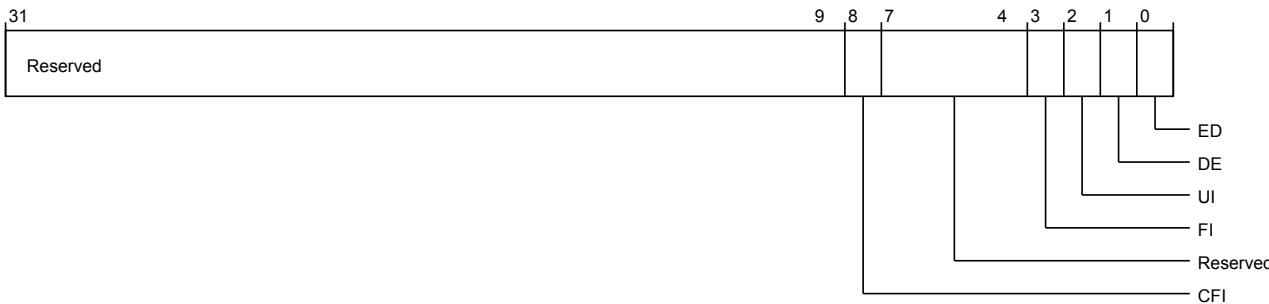


Figure 3-520 por_cxg_ha_por_cxg_ha_errctlr_ns (low)

The following table shows the por_cxg_ha_errctlr_NS lower register bit assignments.

Table 3-540 por_cxg_ha_por_cxg_ha_errctlr_ns (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_cxg_ha_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_cxg_ha_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_cxg_ha_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_cxg_ha_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_cxg_ha_errfr.ED	RW	1'b0

por_cxg_ha_errstatus_NS

Functions as the non-secure error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3110
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

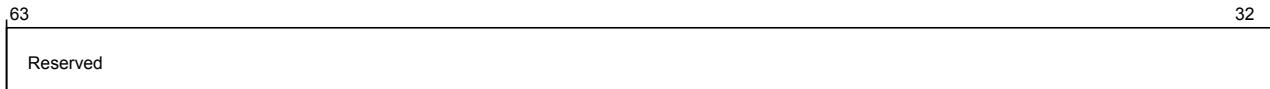


Figure 3-521 por_cxg_ha_por_cxg_ha_errstatus_ns (high)

The following table shows the por_cxg_ha_errstatus_NS higher register bit assignments.

Table 3-541 por_cxg_ha_por_cxg_ha_errstatus_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

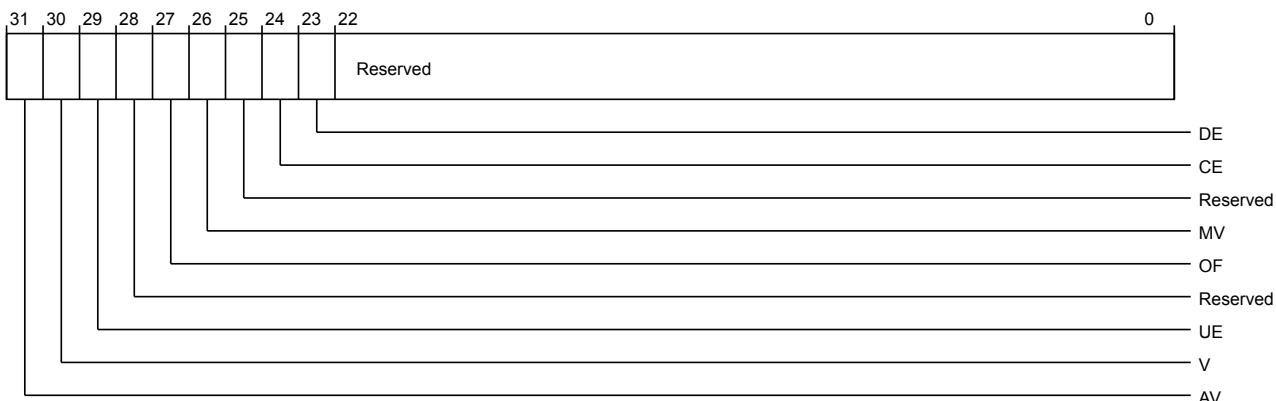


Figure 3-522 por_cxg_ha_por_cxg_ha_errstatus_ns (low)

The following table shows the por_cxg_ha_errstatus_NS lower register bit assignments.

Table 3-542 por_cxg_ha_por_cxg_ha_errstatus_ns (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_cxg_ha_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_cxg_ha_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_cxg_ha_erraddr_NS

Contains the non-secure error record address.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3118
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

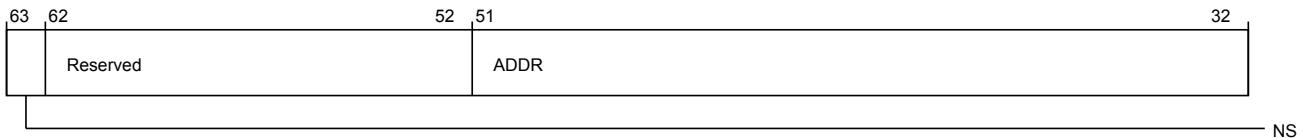


Figure 3-523 por_cxg_ha_por_cxg_ha_erraddr_ns (high)

The following table shows the por_cxg_ha_erraddr_NS higher register bit assignments.

Table 3-543 por_cxg_ha_por_cxg_ha_erraddr_ns (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_cxg_ha_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.

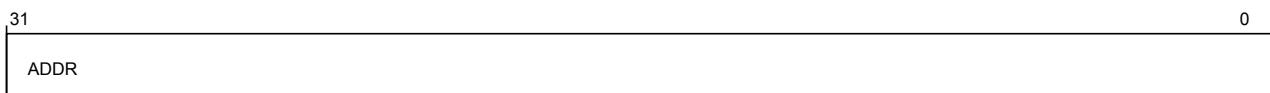


Figure 3-524 por_cxg_ha_por_cxg_ha_erraddr_ns (low)

The following table shows the por_cxg_ha_erraddr_NS lower register bit assignments.

Table 3-544 por_cxg_ha_por_cxg_ha_erraddr_ns (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_cxg_ha_errmisc_NS

Functions as the non-secure miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3120
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

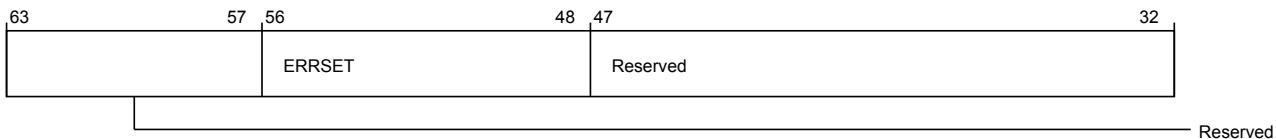


Figure 3-525 por_cxg_ha_por_cxg_ha_errmisc_ns (high)

The following table shows the por_cxg_ha_errmisc_NS higher register bit assignments.

Table 3-545 por_cxg_ha_por_cxg_ha_errmisc_ns (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:48	ERRSET	RAM entry set address for parity error	RW	9'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

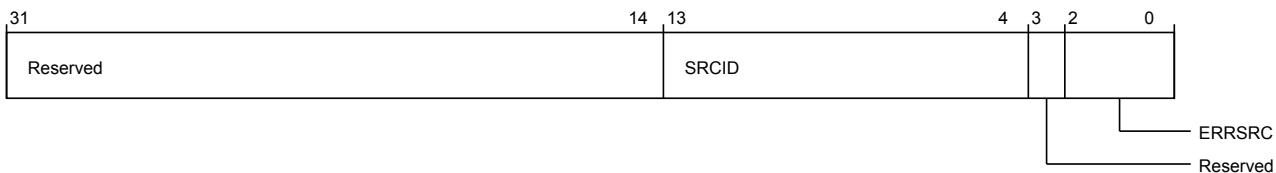


Figure 3-526 por_cxg_ha_por_cxg_ha_errmisc_ns (low)

The following table shows the por_cxg_ha_errmisc_NS lower register bit assignments.

Table 3-546 por_cxg_ha_por_cxg_ha_errmisc_ns (low)

Bits	Field name	Description	Type	Reset
31:14	Reserved	Reserved	RO	-
13:4	SRCID	CCIX RAID of the requestor or the snoop target	RW	10'b0

Table 3-546 por_cxg_ha_por_cxg_ha_errmisc_ns (low) (continued)

Bits	Field name	Description	Type	Reset
3	Reserved	Reserved	RO	-
2:0	ERRSRC	Source of the parity error 3'b000: Read data buffer 0 3'b001: Read data buffer 1 3'b010: Write data buffer 0 3'b011: Write data buffer 1 3'b100: Passive Buffer	RW	3'b000

3.3.10 RN SAM register descriptions

This section lists the RN SAM registers.

por_rnsam_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

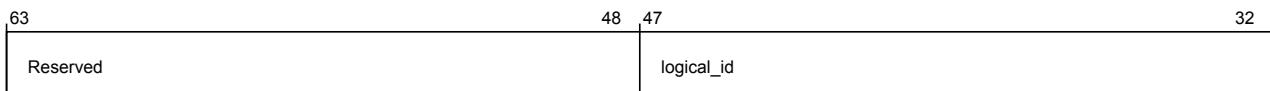


Figure 3-527 por_rnsam_por_rnsam_node_info (high)

The following table shows the por_rnsam_node_info higher register bit assignments.

Table 3-547 por_rnsam_por_rnsam_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID NOTE: RN SAM logical ID is always set to 16'b0.	RO	16'h0

The following image shows the lower register bit assignments.

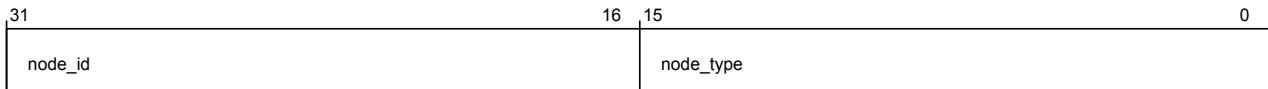


Figure 3-528 por_rnsam_por_rnsam_node_info (low)

The following table shows the por_rnsam_node_info lower register bit assignments.

Table 3-548 por_rnsam_por_rnsam_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h000F

por_rnsam_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

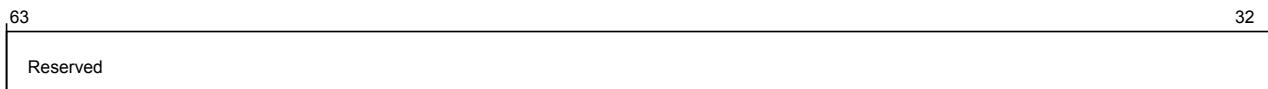


Figure 3-529 por_rnsam_por_rnsam_child_info (high)

The following table shows the por_rnsam_child_info higher register bit assignments.

Table 3-549 por_rnsam_por_rnsam_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

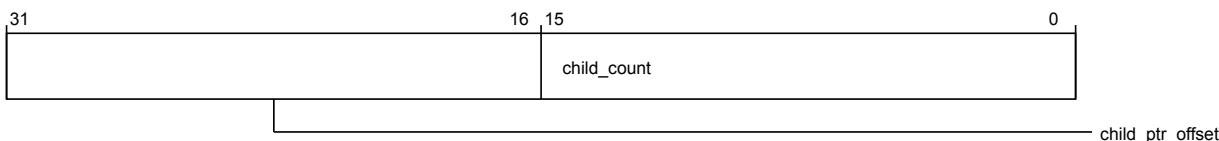


Figure 3-530 por_rnsam_por_rnsam_child_info (low)

The following table shows the por_rnsam_child_info lower register bit assignments.

Table 3-550 por_rnsam_por_rnsam_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_rnsam_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

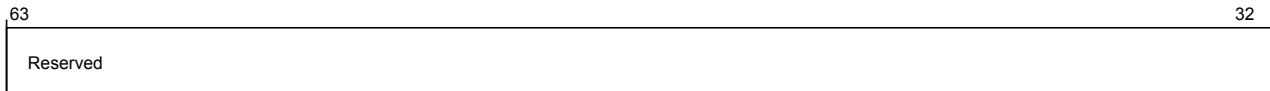


Figure 3-531 por_rnsam_por_rnsam_secure_register_groups_override (high)

The following table shows the por_rnsam_secure_register_groups_override higher register bit assignments.

Table 3-551 por_rnsam_por_rnsam_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

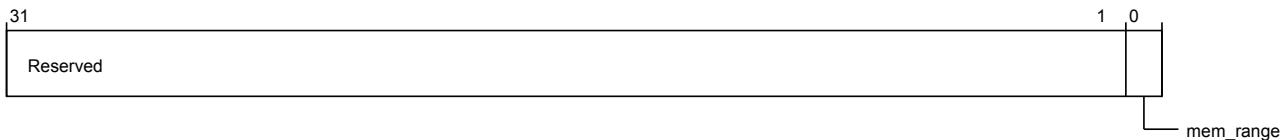


Figure 3-532 por_rnsam_por_rnsam_secure_register_groups_override (low)

The following table shows the por_rnsam_secure_register_groups_override lower register bit assignments.

Table 3-552 por_rnsam_por_rnsam_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	mem_range	Allows non-secure access to secure mem_ranges registers	RW	1'b0

por_rnsam_unit_info

Provides component identification information for RN SAM.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

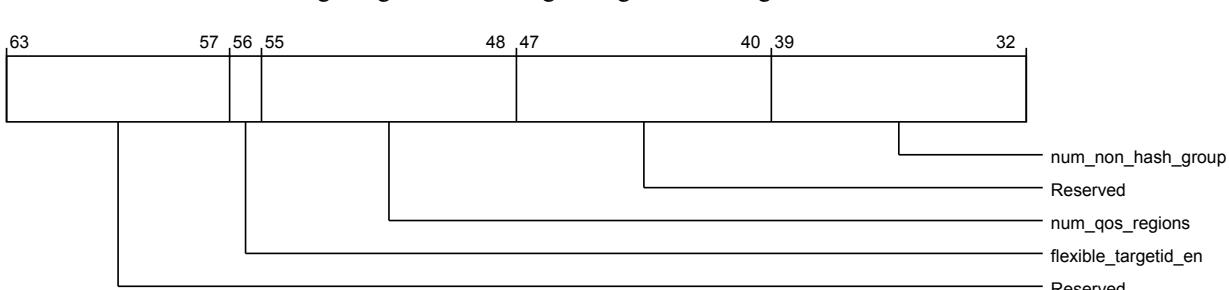


Figure 3-533 por rnsam por rnsam unit info (high)

The following table shows the por_rnsam unit info higher register bit assignments.

Table 3-553 por rnsam por rnsam unit info (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56	flexible_targetid_en	flexible target enable to preserve backward compatibility	RO	Configuration dependent
55:48	num_qos_regions	Number of QOS regions	RO	Configuration dependent
47:40	Reserved	Reserved	RO	-
39:32	num_non_hash_group	Number of non-hashed groups supported	RO	Configuration dependent

The following image shows the lower register bit assignments.

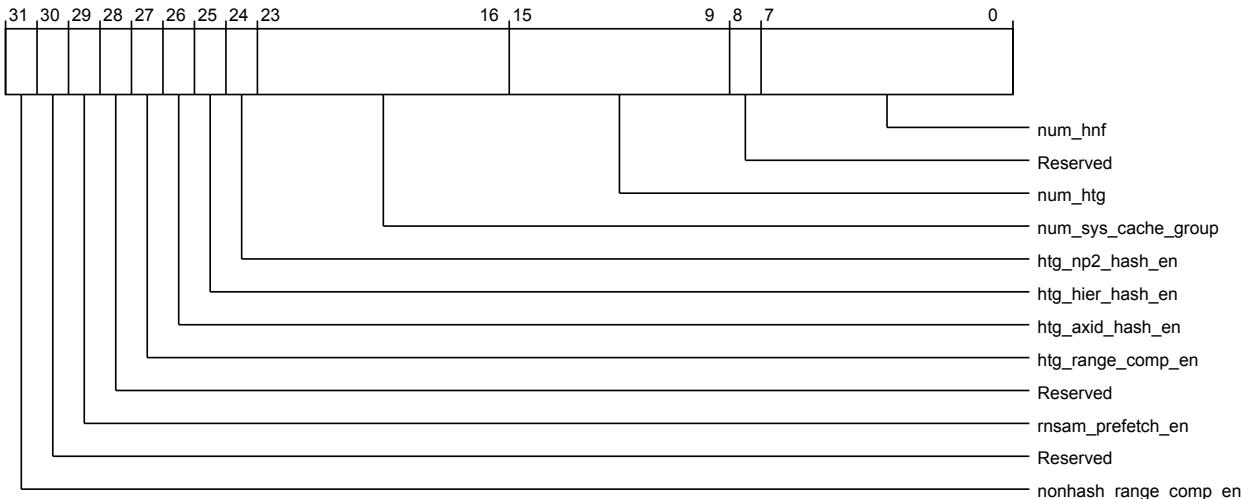


Figure 3-534 por_rnsam_por_rnsam_unit_info (low)

The following table shows the por_rnsam unit info lower register bit assignments.

Table 3-554 por_rnsam_por_rnsam_unit_info (low)

Bits	Field name	Description	Type	Reset
31	nonhash_range_comp_en	Define start and end address for each HTG region	RO	Configuration dependent
30	Reserved	Reserved	RO	-
29	rnsam_prefetch_en	RNSAM prefetch enabled	RO	Configuration dependent
28	Reserved	Reserved	RO	-
27	htg_range_comp_en	Define start and end address for each HTG region	RO	Configuration dependent
26	htg_axid_hash_en	Enable AXID based hashing scheme	RO	Configuration dependent
25	htg_hier_hash_en	Enable Hierarchical hashing scheme	RO	Configuration dependent
24	htg_np2_hash_en	Enable non-power of two hash scheme	RO	Configuration dependent
23:16	num_sys_cache_group	Number of system cache groups supported	RO	Configuration dependent
15:9	num_htg	Number of Hashed target groups	RO	Configuration dependent
8	Reserved	Reserved	RO	-
7:0	num_hnf	Number of hashed targets supported	RO	Configuration dependent

por_rnsam_unit_info1

Provides component identification information for RN SAM.

Its characteristics are:

Type RO

Register width (Bits) 64

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

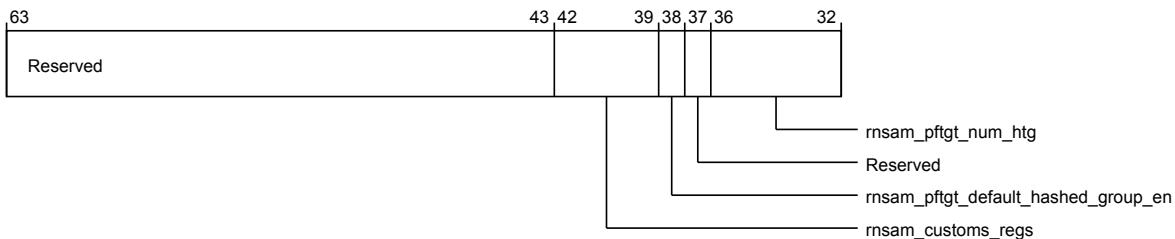


Figure 3-535 por_rnsam_por_rnsam_unit_info1 (high)

The following table shows the por_rnsam_unit_info1 higher register bit assignments.

Table 3-555 por_rnsam_por_rnsam_unit_info1 (high)

Bits	Field name	Description	Type	Reset
63:43	Reserved	Reserved	RO	-
42:39	rnsam_customs_regs	Number of customer specific registers for customer implemented logic	RO	Configuration dependent
38	rnsam_pftgt_default_hashed_group_en	Enable default hashed group for prefetch transactions. To support backward compatible, set this parameter	RO	Configuration dependent
37	Reserved	Reserved	RO	-
36:32	rnsam_pftgt_num_htg	Number of prefetch HTG regions supported per System Cache Group by the RNSAM	RO	Configuration dependent

The following image shows the lower register bit assignments.

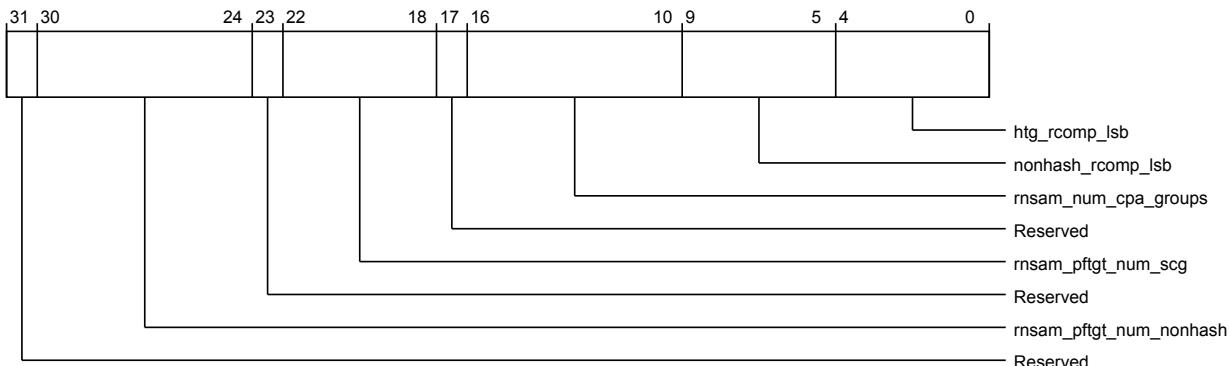


Figure 3-536 por_rnsam_por_rnsam_unit_info1 (low)

The following table shows the por_rnsam_unit_info1 lower register bit assignments.

Table 3-556 por_rnsam_por_rnsam_unit_info1 (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:24	rnsam_pftgt_num_nonhash	Number of prefetch non-hashed regions supported per System Cache Group by the RNSAM	RO	Configuration dependent
23	Reserved	Reserved	RO	-
22:18	rnsam_pftgt_num_scg	Number of system cache groups enabled for prefetch targets	RO	Configuration dependent
17	Reserved	Reserved	RO	-
16:10	rnsam_num_cpa_groups	Number of CPA groups	RO	Configuration dependent
9:5	nonhash_rcomp_lsb	NONHASH RCOMP LSB bit position defining minimum region size	RO	Configuration dependent
4:0	htg_rcomp_lsb	HTG RCOMP LSB bit position defining minimum region size	RO	Configuration dependent

non_hash_mem_region_reg_24-63

This register repeats 39 times. It parametrized by the index from 24 to 63. Configures non-hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

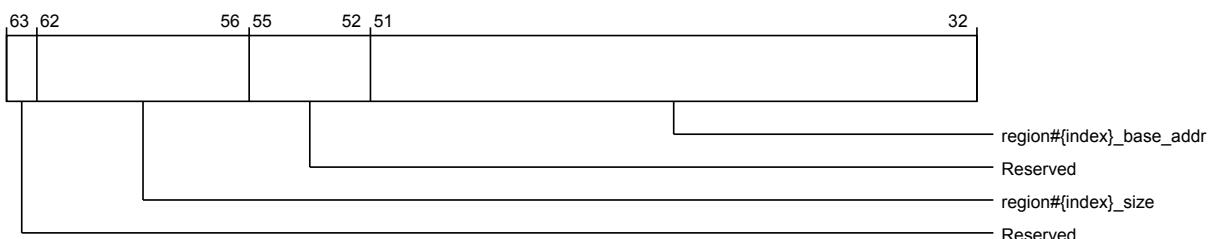


Figure 3-537 por_rnsam_non_hash_mem_region_reg_24-63 (high)

The following table shows the non_hash_mem_region_reg_24-63 higher register bit assignments.

Table 3-557 por_rnsam_non_hash_mem_region_reg_24-63 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	region#{index}_size	<p>Memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	7'h0
55:52	Reserved	Reserved	RO	-
51:32	region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

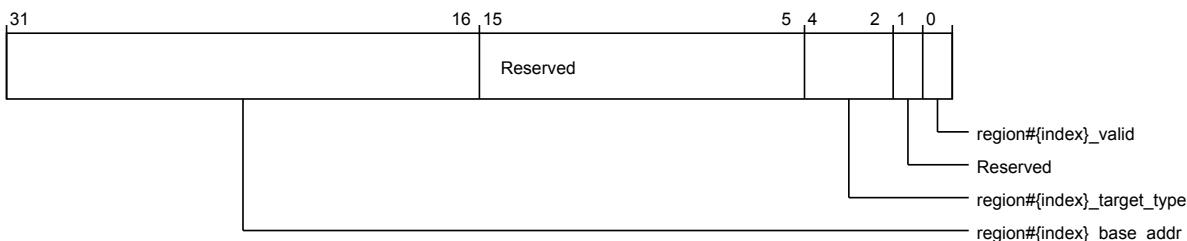


Figure 3-538 por_rnsam_non_hash_mem_region_reg_24-63 (low)

The following table shows the non_hash_mem_region_reg_24-63 lower register bit assignments.

Table 3-558 por_rnsam_non_hash_mem_region_reg_24-63 (low)

Bits	Field name	Description	Type	Reset
31:16	region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-
4:2	region#{index}_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000
1	Reserved	Reserved	RO	-
0	region#{index}_valid	<p>Memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

non_hash_mem_region_reg_24-63

This register repeats 39 times. It parametrized by the index from 24 to 63. Configures non-hashed memory regions

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset	16'hC00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

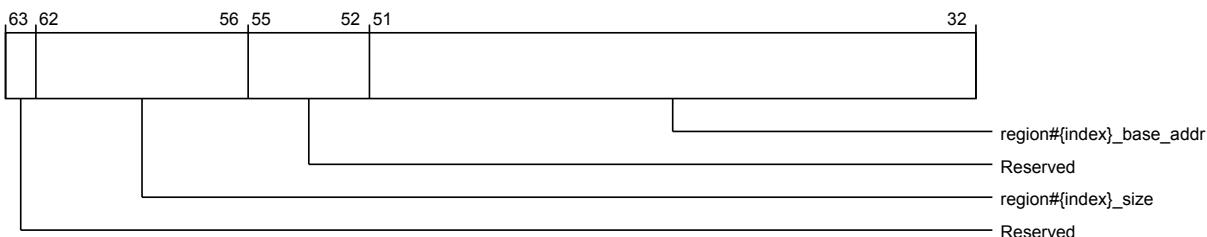


Figure 3-539 por_rnsam_non_hash_mem_region_reg_24-63 (high)

The following table shows the non_hash_mem_region_reg_24-63 higher register bit assignments.

Table 3-559 por_rnsam_non_hash_mem_region_reg_24-63 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	region#{index}_size	<p>Memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	7'h0
55:52	Reserved	Reserved	RO	-
51:32	region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

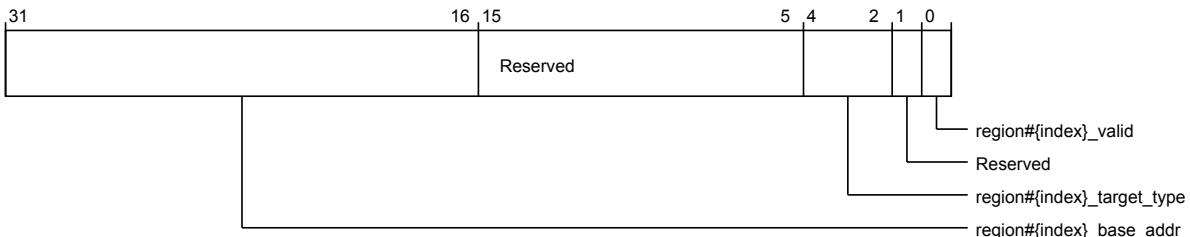


Figure 3-540 por_rnsam_non_hash_mem_region_reg_24-63 (low)

The following table shows the non_hash_mem_region_reg_24-63 lower register bit assignments.

Table 3-560 por_rnsam_non_hash_mem_region_reg_24-63 (low)

Bits	Field name	Description	Type	Reset
31:16	region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-
4:2	region#{index}_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000
1	Reserved	Reserved	RO	-
0	region#{index}_valid	<p>Memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

non_hash_mem_region_cfg2_reg_24-63

This register repeats 39 times. It parametrized by the index from 24 to 63. Configures non-hashed memory region end address

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hCC0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

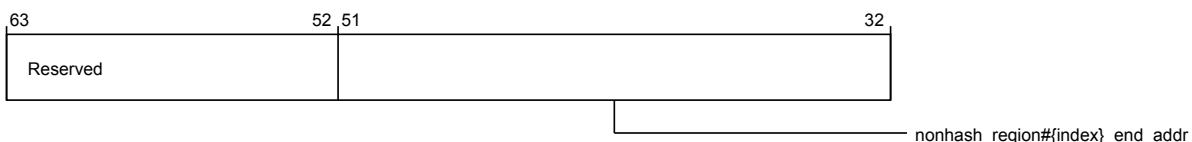


Figure 3-541 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (high)

The following table shows the non_hash_mem_region_cfg2_reg_24-63 higher register bit assignments.

Table 3-561 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	nonhash_region#{index}_end_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

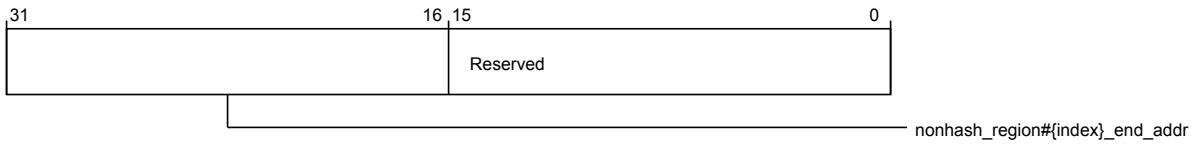


Figure 3-542 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (low)

The following table shows the non_hash_mem_region_cfg2_reg_24-63 lower register bit assignments.

Table 3-562 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (low)

Bits	Field name	Description	Type	Reset
31:16	nonhash_region#{index}_end_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0
15:0	Reserved	Reserved	RO	-

non_hash_mem_region_cfg2_reg_24-63

This register repeats 39 times. It parametrized by the index from 24 to 63. Configures non-hashed memory region end address

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hCC0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

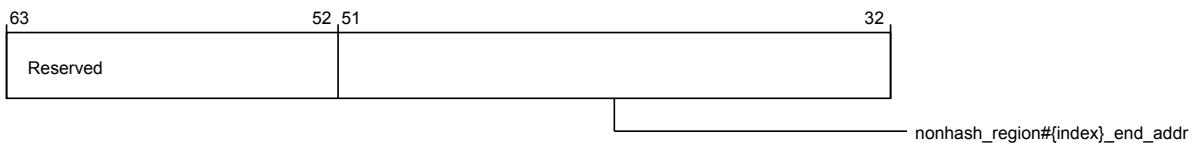


Figure 3-543 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (high)

The following table shows the non_hash_mem_region_cfg2_reg_24-63 higher register bit assignments.

Table 3-563 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	nonhash_region#{index}_end_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

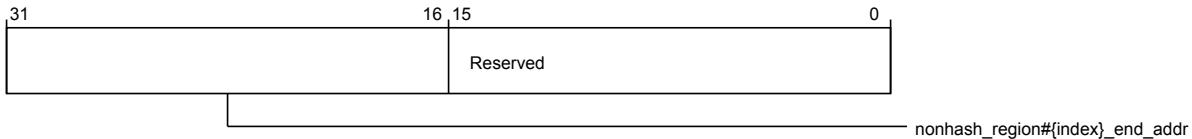


Figure 3-544 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (low)

The following table shows the non_hash_mem_region_cfg2_reg_24-63 lower register bit assignments.

Table 3-564 por_rnsam_non_hash_mem_region_cfg2_reg_24-63 (low)

Bits	Field name	Description	Type	Reset
31:16	nonhash_region#{index}_end_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_NONHASH_RCOMP_LSB_PARAM</p>	RW	36'h0
15:0	Reserved	Reserved	RO	-

non_hash_tgt_nodeid_16-15

This register repeats -1 times. It parametrized by the index from 16 to 15. Configures non-hashed target node IDs # $\{4 * \text{index}\}$ to # $\{4 * \text{index} + 3\}$.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD80
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

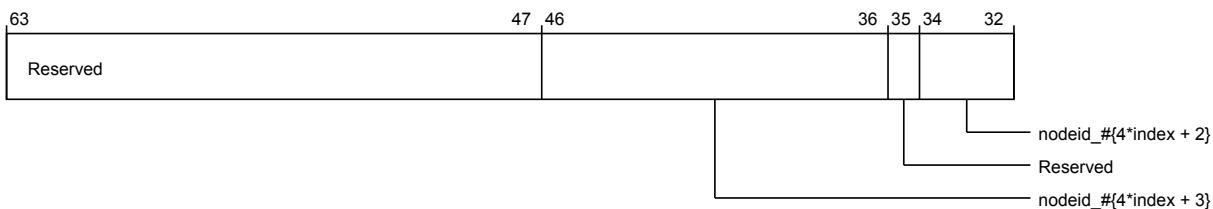


Figure 3-545 por_rnsam_non_hash_tgt_nodeid_16-15 (high)

The following table shows the non_hash_tgt_nodeid_16-15 higher register bit assignments.

Table 3-565 por_rnsam_non_hash_tgt_nodeid_16-15 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	nodeid_#\{4*index + 3\}	<p>Non-hashed target node ID # $\{4 * \text{index} + 3\}$ </p>	RW	11'b000000000000

Table 3-565 por_rnsam_non_hash_tgt_nodeid_16-15 (high) (continued)

Bits	Field name	Description	Type	Reset
35	Reserved	Reserved	RO	-
34:32	nodeid_{#{4*index + 2}}	<p>Non-hashed target node ID #{4*index + 2}</p>	RW	11'b000000000000

The following image shows the lower register bit assignments.

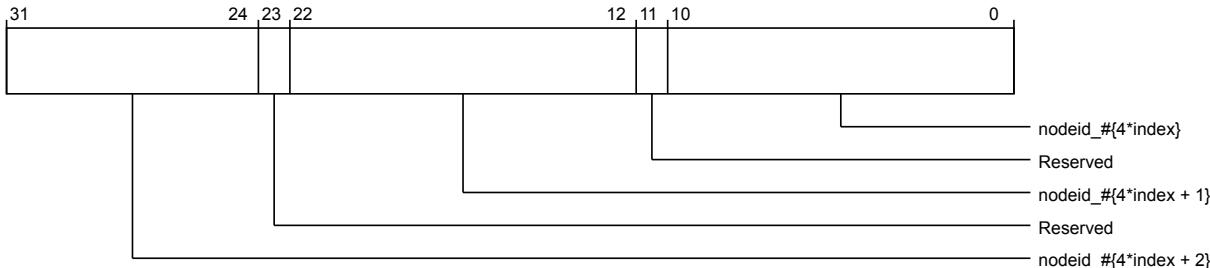


Figure 3-546 por_rnsam_non_hash_tgt_nodeid_16-15 (low)

The following table shows the non_hash_tgt_nodeid_16-15 lower register bit assignments.

Table 3-566 por_rnsam_non_hash_tgt_nodeid_16-15 (low)

Bits	Field name	Description	Type	Reset
31:24	nodeid_{#{4*index + 2}}	<p>Non-hashed target node ID #{4*index + 2}</p>	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	nodeid_{#{4*index + 1}}	<p>Non-hashed target node ID #{4*index + 1}</p>	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	nodeid_{#{4*index}}	<p>Non-hashed target node ID #{4*index}</p>	RW	11'b000000000000

non_hash_tgt_nodeid_16-15

This register repeats -1 times. It parametrized by the index from 16 to 15. Configures non-hashed target node IDs #{4*index} to #{4*index + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD80
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

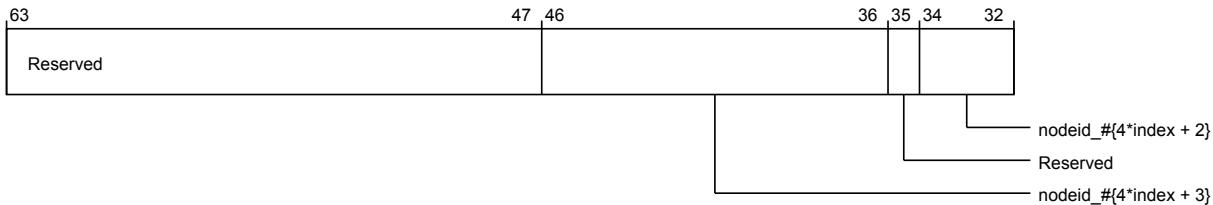


Figure 3-547 por_rnsam_non_hash_tgt_nodeid_16-15 (high)

The following table shows the non_hash_tgt_nodeid_16-15 higher register bit assignments.

Table 3-567 por_rnsam_non_hash_tgt_nodeid_16-15 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	nodeid_{4*index + 3}	<p>Non-hashed target node ID #{4*index + 3}</p>	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	nodeid_{4*index + 2}	<p>Non-hashed target node ID #{4*index + 2}</p>	RW	11'b000000000000

The following image shows the lower register bit assignments.

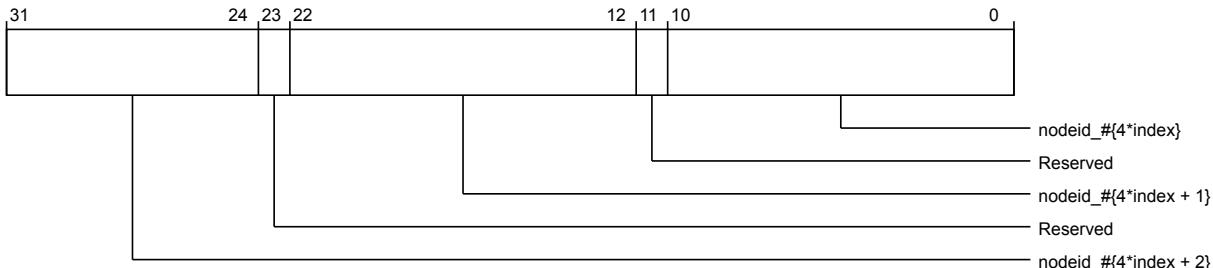


Figure 3-548 por_rnsam_non_hash_tgt_nodeid_16-15 (low)

The following table shows the non_hash_tgt_nodeid_16-15 lower register bit assignments.

Table 3-568 por_rnsam_non_hash_tgt_nodeid_16-15 (low)

Bits	Field name	Description	Type	Reset
31:24	nodeid_{4*index + 2}	<p>Non-hashed target node ID #{4*index + 2}</p>	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	nodeid_{4*index + 1}	<p>Non-hashed target node ID #{4*index + 1}</p>	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	nodeid_{4*index}	<p>Non-hashed target node ID #{4*index}</p>	RW	11'b000000000000

cml_port_aggr_mode_ctrl_reg

Configures the CCIX port aggregation modes for all non-hashed memory regions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h11A0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

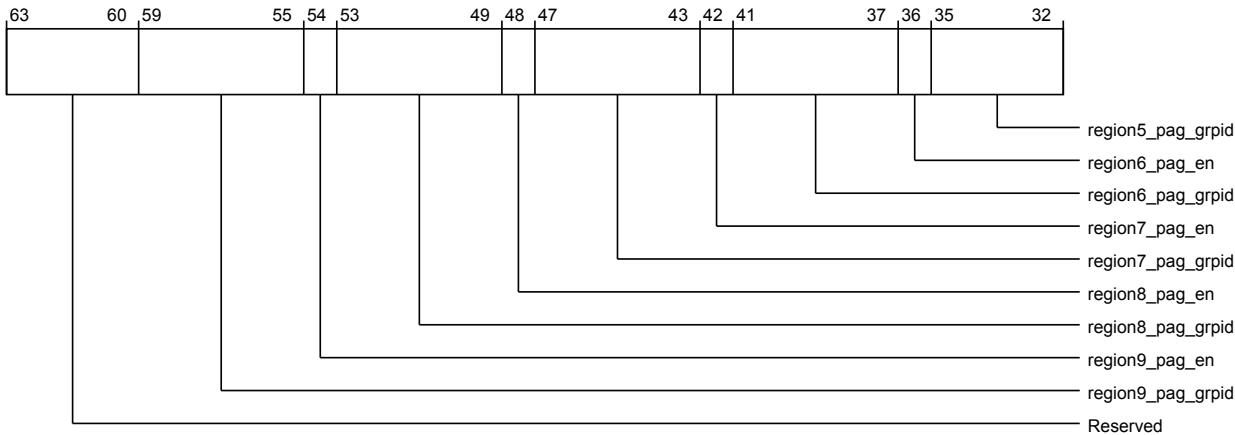


Figure 3-549 por_rnsam_cml_port_aggr_mode_ctrl_reg (high)

The following table shows the cml_port_aggr_mode_ctrl_reg higher register bit assignments.

Table 3-569 por_rnsam_cml_port_aggr_mode_ctrl_reg (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:55	region9_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
54	region9_pag_en	Enables the CPA mode for non-hashed memory region 9	RW	1'b0
53:49	region8_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
48	region8_pag_en	Enables the CPA mode for non-hashed memory region 8	RW	1'b0
47:43	region7_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
42	region7_pag_en	Enables the CPA mode for non-hashed memory region 7	RW	1'b0
41:37	region6_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
36	region6_pag_en	Enables the CPA mode for non-hashed memory region 6	RW	1'b0
35:32	region5_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0

The following image shows the lower register bit assignments.

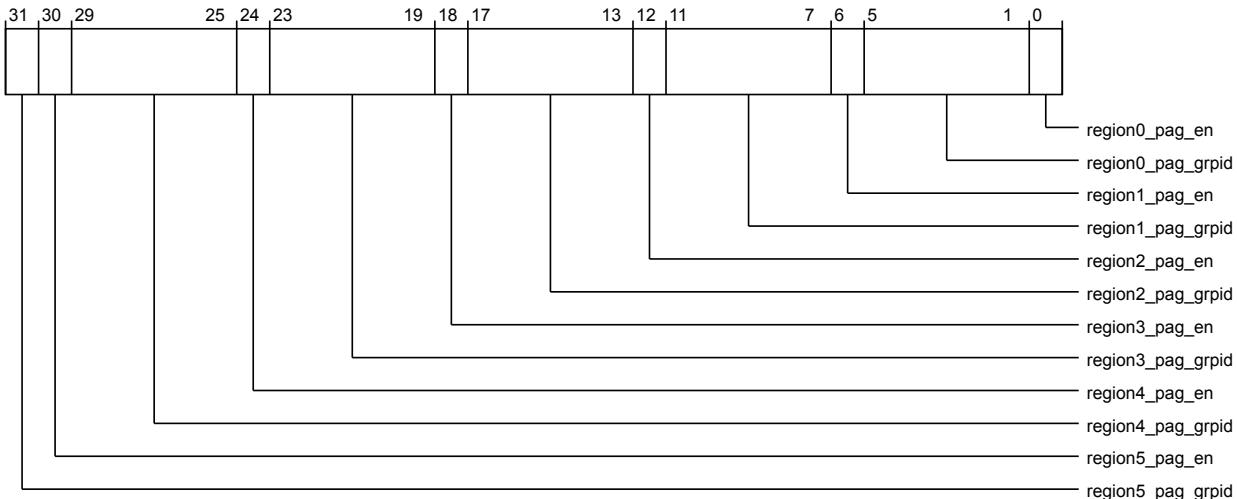


Figure 3-550 por_rnsam_cml_port_aggr_mode_ctrl_reg (low)

The following table shows the cml_port_aggr_mode_ctrl_reg lower register bit assignments.

Table 3-570 por_rnsam_cml_port_aggr_mode_ctrl_reg (low)

Bits	Field name	Description	Type	Reset
31	region5_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
30	region5_pag_en	Enables the CPA mode for non-hashed memory region 5	RW	1'b0
29:25	region4_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
24	region4_pag_en	Enables the CPA mode for non-hashed memory region 4	RW	1'b0
23:19	region3_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
18	region3_pag_en	Enables the CPA mode for non-hashed memory region 3	RW	1'b0
17:13	region2_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
12	region2_pag_en	Enables the CPA mode for non-hashed memory region 2	RW	1'b0
11:7	region1_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
6	region1_pag_en	Enables the CPA mode for non-hashed memory region 1	RW	1'b0
5:1	region0_pag_grpid	Specifies CCIX port aggregation group ID	RW	5'h0
0	region0_pag_en	Enables the CPA mode for non-hashed memory region 0	RW	1'b0

cml_port_aggr_mode_ctrl_reg_4-6

This register repeats 2 times. It parametrized by the index from 4 to 6. Configures the CCIX port aggregation modes for all non-hashed memory regions.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset	16'h11A0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-551 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (high)

The following table shows the cml_port_aggr_mode_ctrl_reg_4-6 higher register bit assignments.

Table 3-571 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:55	region#{index*10 + 9}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
54	region#{index*10 + 9}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 9}</p>	RW	1'b0
53:49	region#{index*10 + 8}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
48	region#{index*10 + 8}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 8}</p>	RW	1'b0
47:43	region#{index*10 + 7}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
42	region#{index*10 + 7}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 7}</p>	RW	1'b0
41:37	region#{index*10 + 6}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
36	region#{index*10 + 6}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 6}</p>	RW	1'b0
35:32	region#{index*10 + 5}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0

The following image shows the lower register bit assignments.

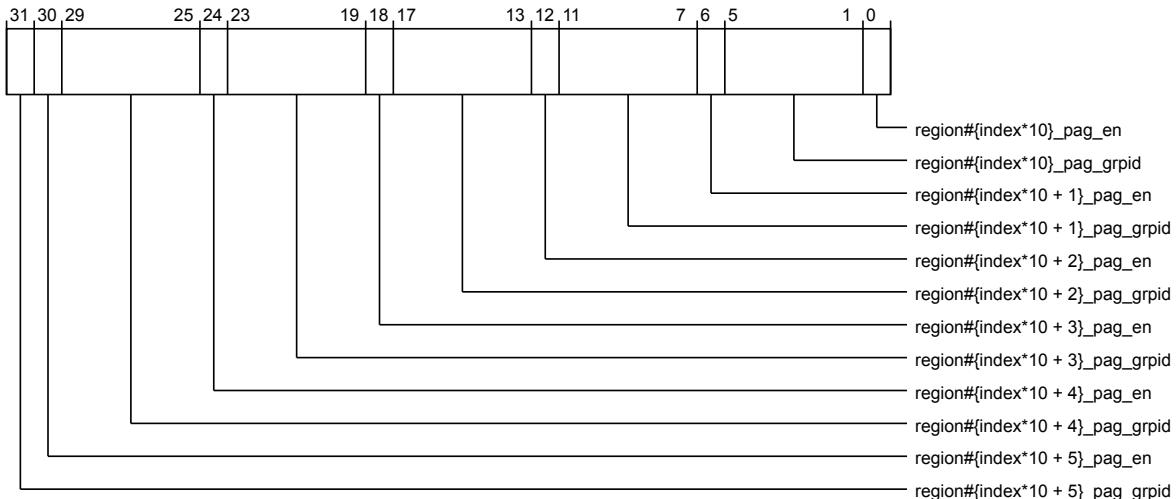


Figure 3-552 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (low)

The following table shows the cml_port_aggr_mode_ctrl_reg_4-6 lower register bit assignments.

Table 3-572 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (low)

Bits	Field name	Description	Type	Reset
31	region#{index*10 + 5}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
30	region#{index*10 + 5}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 5}</p>	RW	1'b0
29:25	region#{index*10 + 4}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
24	region#{index*10 + 4}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 4}</p>	RW	1'b0
23:19	region#{index*10 + 3}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
18	region#{index*10 + 3}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 3}</p>	RW	1'b0
17:13	region#{index*10 + 2}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
12	region#{index*10 + 2}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 2}</p>	RW	1'b0
11:7	region#{index*10 + 1}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
6	region#{index*10 + 1}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 1}</p>	RW	1'b0
5:1	region#{index*10}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
0	region#{index*10}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10}</p>	RW	1'b0

cml_port_aggr_mode_ctrl_reg_4-6

This register repeats 2 times. It parametrized by the index from 4 to 6. Configures the CCIX port aggregation modes for all non-hashed memory regions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h11A0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

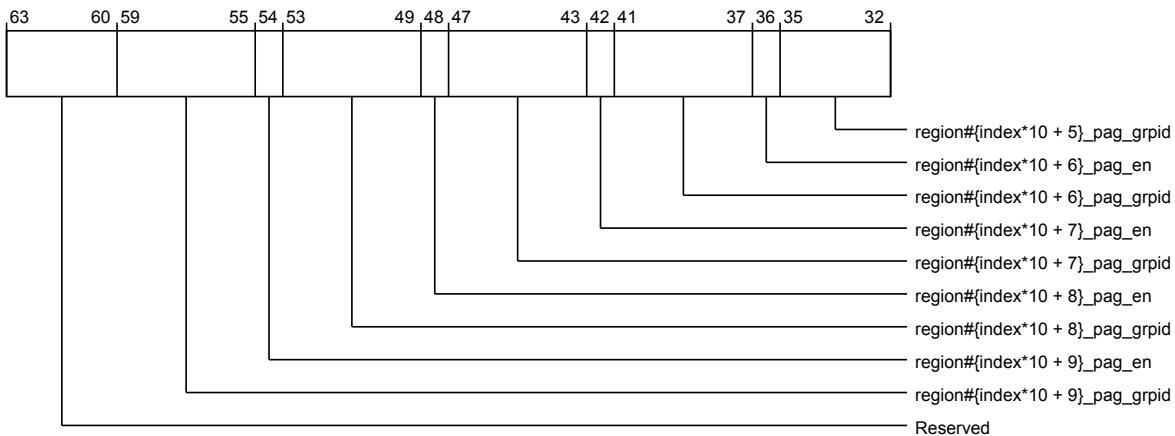


Figure 3-553 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (high)

The following table shows the cml_port_aggr_mode_ctrl_reg_4-6 higher register bit assignments.

Table 3-573 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:55	region#{index*10 + 9}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
54	region#{index*10 + 9}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 9}</p>	RW	1'b0
53:49	region#{index*10 + 8}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
48	region#{index*10 + 8}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 8}</p>	RW	1'b0
47:43	region#{index*10 + 7}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
42	region#{index*10 + 7}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 7}</p>	RW	1'b0
41:37	region#{index*10 + 6}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0

Table 3-573 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (high) (continued)

Bits	Field name	Description	Type	Reset
36	region#{index*10 + 6}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 6}</p>	RW	1'b0
35:32	region#{index*10 + 5}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0

The following image shows the lower register bit assignments.

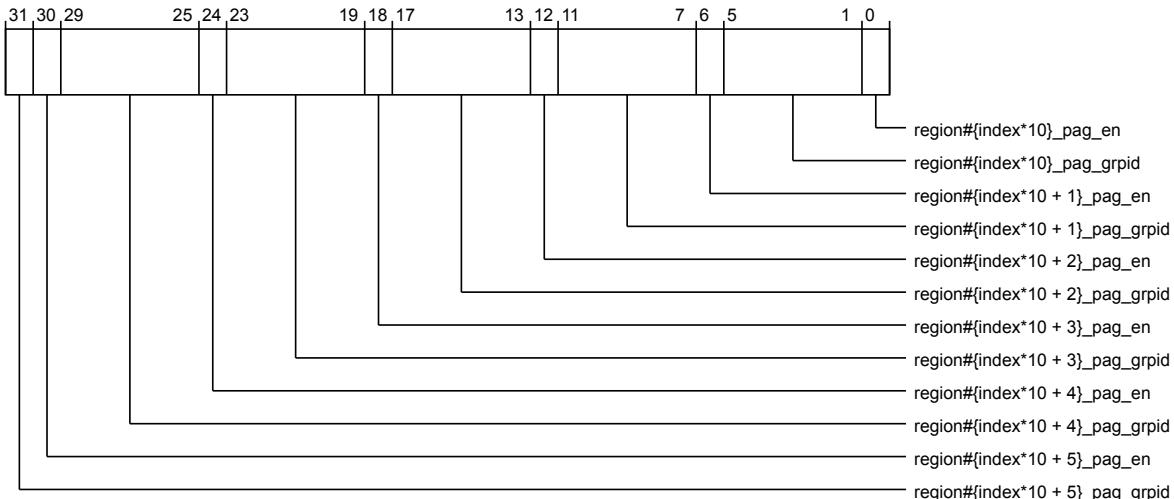


Figure 3-554 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (low)

The following table shows the cml_port_aggr_mode_ctrl_reg_4-6 lower register bit assignments.

Table 3-574 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (low)

Bits	Field name	Description	Type	Reset
31	region#{index*10 + 5}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
30	region#{index*10 + 5}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 5}</p>	RW	1'b0
29:25	region#{index*10 + 4}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
24	region#{index*10 + 4}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 4}</p>	RW	1'b0
23:19	region#{index*10 + 3}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
18	region#{index*10 + 3}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 3}</p>	RW	1'b0
17:13	region#{index*10 + 2}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
12	region#{index*10 + 2}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 2}</p>	RW	1'b0
11:7	region#{index*10 + 1}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0

Table 3-574 por_rnsam_cml_port_aggr_mode_ctrl_reg_4-6 (low) (continued)

Bits	Field name	Description	Type	Reset
6	region#{index*10 + 1}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10 + 1}</p>	RW	1'b0
5:1	region#{index*10}_pag_grpid	<p>Specifies CCIX port aggregation group ID</p>	RW	5'h0
0	region#{index*10}_pag_en	<p>Enables the CPA mode for non-hashed memory region #{index*10}</p>	RW	1'b0

sys_cache_grp_region_0-3

This register repeats 3 times. It parametrized by the index from 0 to 3. Configures hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE00 + #{8*[0, 1, 2, 3]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

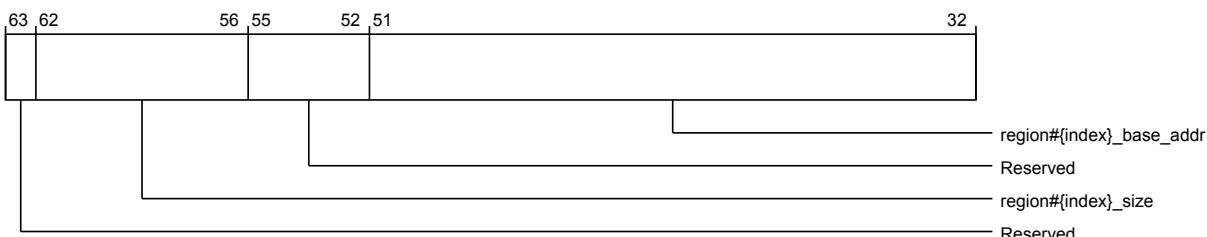


Figure 3-555 por_rnsam_sys_cache_grp_region_0-3 (high)

The following table shows the sys_cache_grp_region_0-3 higher register bit assignments.

Table 3-575 por_rnsam_sys_cache_grp_region_0-3 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	region#{index}_size	Memory region #{index} size CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).	RW	7'b0000000

Table 3-575 por_rnsam_sys_cache_grp_region_0-3 (high) (continued)

Bits	Field name	Description	Type	Reset
55:52	Reserved	Reserved	RO	-
51:32	region#{index}_base_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_HTG_RCOMP_LSB_PARAM	RW	36'b00000000000000000000000000000000

The following image shows the lower register bit assignments.

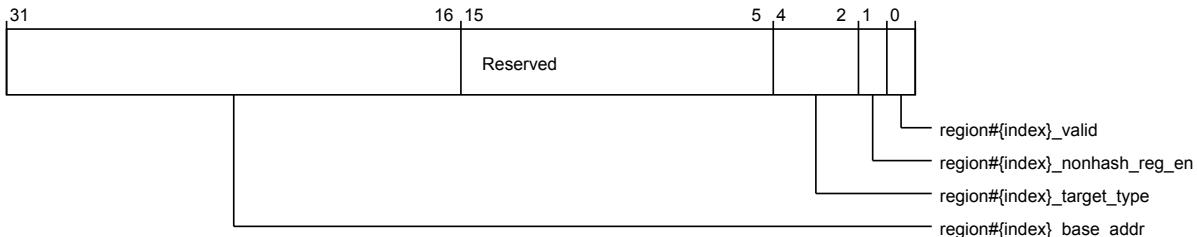


Figure 3-556 por_rnsam_sys_cache_grp_region_0-3 (low)

The following table shows the sys_cache_grp_region_0-3 lower register bit assignments.

Table 3-576 por_rnsam_sys_cache_grp_region_0-3 (low)

Bits	Field name	Description	Type	Reset
31:16	region#{index}_base_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_HTG_RCOMP_LSB_PARAM	RW	36'b00000000000000000000000000000000
15:5	Reserved	Reserved	RO	-
4:2	region#{index}_target_type	Indicates node type 3'b000: HN-F 3'b001: HN-I 3'b010: CXRA 3'b011: HN-P 3'b100: PCI-CXRA Others: Reserved CONSTRAINT: Only applicable for RN-I	RW	3'b000
1	region#{index}_nonhash_reg_en	Enables hashed region #{index} to select non-hashed node	RW	1'b0
0	region#{index}_valid	Memory region #{index} valid 1'b0: Not valid 1'b1: Valid for memory region comparison	RW	1'b0

hashed_tgt_grp_cfg1_region_8-31

This register repeats 23 times. It parametrized by the index from 8 to 31. Configures hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

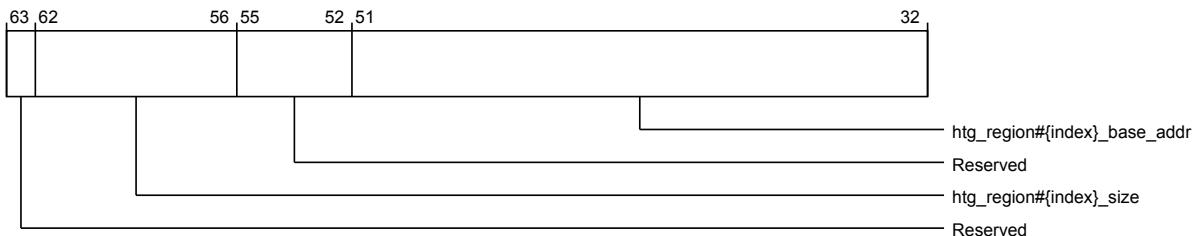


Figure 3-557 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (high)

The following table shows the hashed_tgt_grp_cfg1_region_8-31 higher register bit assignments.

Table 3-577 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	htg_region#{index}_size	<p>Memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	7'b0000000
55:52	Reserved	Reserved	RO	-
51:32	htg_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_HTG_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

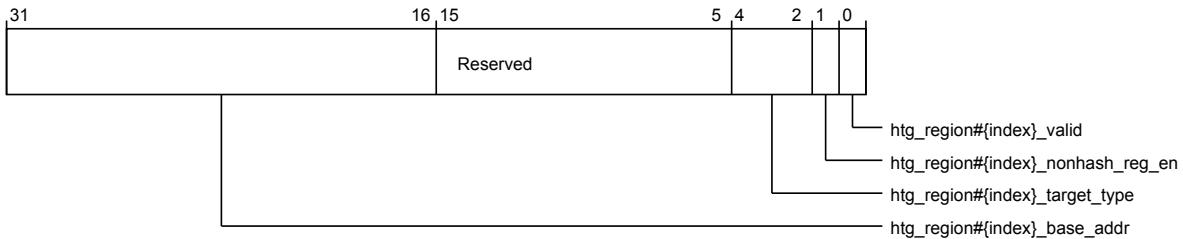


Figure 3-558 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (low)

The following table shows the hashed_tgt_grp_cfg1_region_8-31 lower register bit assignments.

Table 3-578 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (low)

Bits	Field name	Description	Type	Reset
31:16	htg_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-
4:2	htg_region#{index}_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000
1	htg_region#{index}_nonhash_reg_en	<p>Enables hashed region #{index} to select non-hashed node</p>	RW	1'b0
0	htg_region#{index}_valid	<p>Memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

hashed_tgt_grp_cfg1_region_8-31

This register repeats 23 times. It parametrized by the index from 8 to 31. Configures hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

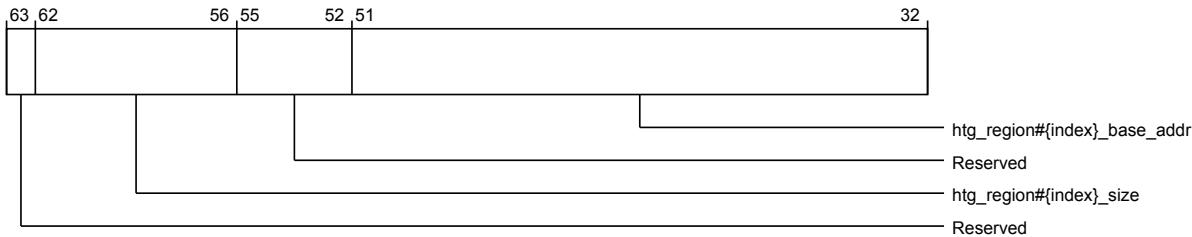


Figure 3-559 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (high)

The following table shows the hashed_tgt_grp_cfg1_region_8-31 higher register bit assignments.

Table 3-579 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	htg_region#{index}_size	<p>Memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	7'b0000000
55:52	Reserved	Reserved	RO	-
51:32	htg_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

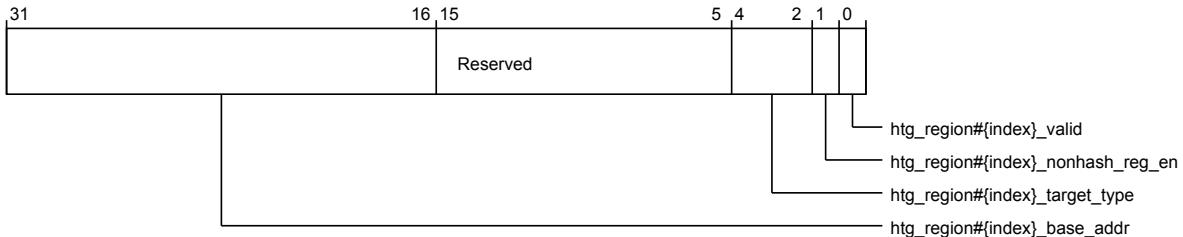


Figure 3-560 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (low)

The following table shows the hashed_tgt_grp_cfg1_region_8-31 lower register bit assignments.

Table 3-580 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (low)

Bits	Field name	Description	Type	Reset
31:16	htg_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-
4:2	htg_region#{index}_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000

Table 3-580 por_rnsam_hashed_tgt_grp_cfg1_region_8-31 (low) (continued)

Bits	Field name	Description	Type	Reset
1	htg_region#{index}_nonhash_reg_en	<p>Enables hashed region #{index} to select non-hashed node</p>	RW	1'b0
0	htg_region#{index}_valid	<p>Memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

hashed_tgt_grp_cfg2_region_0-31

This register repeats 31 times. It parametrized by the index from 0 to 31. Configures hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3100 + #{8 * [0, 1, 2, .., 30, 31]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

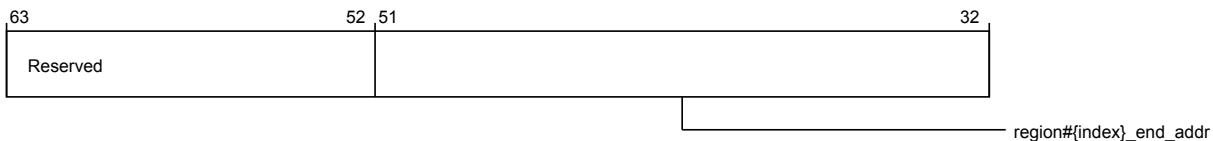


Figure 3-561 por_rnsam_hashed_tgt_grp_cfg2_region_0-31 (high)

The following table shows the hashed_tgt_grp_cfg2_region_0-31 higher register bit assignments.

Table 3-581 por_rnsam_hashed_tgt_grp_cfg2_region_0-31 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	region#{index}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM	RW	36'h0

The following image shows the lower register bit assignments.

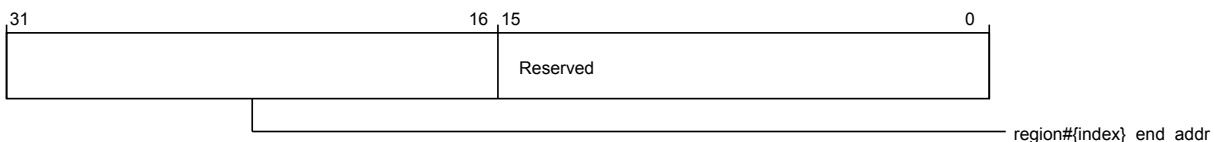


Figure 3-562 por_rnsam_hashed_tgt_grp_cfg2_region_0-31 (low)

The following table shows the hashed_tgt_grp_cfg2_region_0-31 lower register bit assignments.

Table 3-582 por_rnsam_hashed_tgt_grp_cfg2_region_0-31 (low)

Bits	Field name	Description	Type	Reset
31:16	region#{index}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM	RW	36'h0
15:0	Reserved	Reserved	RO	-

sys_cache_grp_secondary_reg_8-3

This register repeats -5 times. It parametrized by the index from 8 to 3. Configures secondary hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

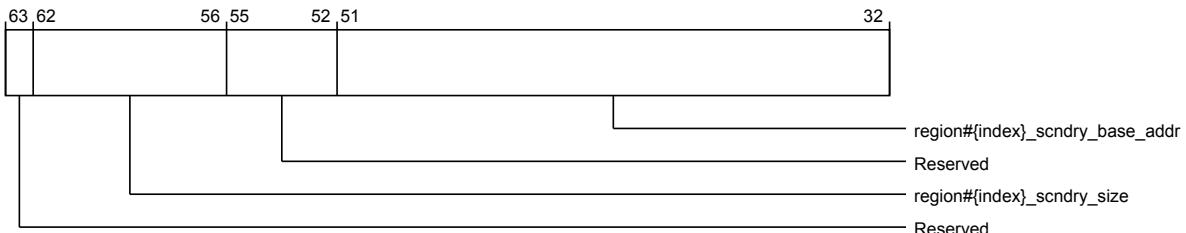


Figure 3-563 por_rnsam_sys_cache_grp_secondary_reg_8-3 (high)

The following table shows the sys_cache_grp_secondary_reg_8-3 higher register bit assignments.

Table 3-583 por_rnsam_sys_cache_grp_secondary_reg_8-3 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	region#{index}_scndry_size	<p>Secondary memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	5'b00000
55:52	Reserved	Reserved	RO	-
51:32	region#{index}_scndry_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

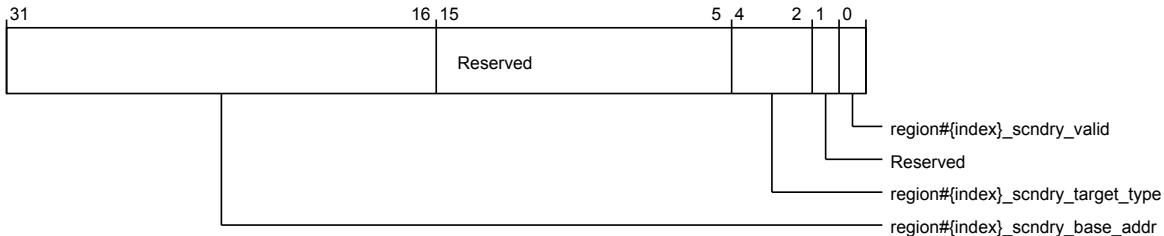


Figure 3-564 por_rnsam_sys_cache_grp_secondary_reg_8-3 (low)

The following table shows the sys_cache_grp_secondary_reg_8-3 lower register bit assignments.

Table 3-584 por_rnsam_sys_cache_grp_secondary_reg_8-3 (low)

Bits	Field name	Description	Type	Reset
31:16	region#{index}_scndry_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_HTG_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-
4:2	region#{index}_scndry_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000
1	Reserved	Reserved	RO	-
0	region#{index}_scndry_valid	<p>Secondary memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

sys_cache_grp_secondary_reg_8-3

This register repeats -5 times. It parametrized by the index from 8 to 3. Configures secondary hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

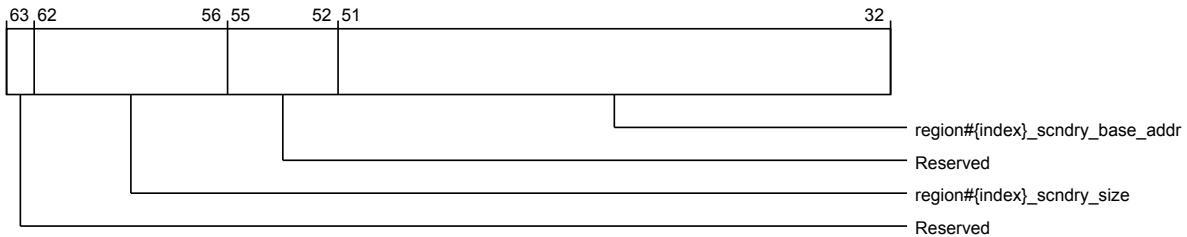


Figure 3-565 por_rnsam_sys_cache_grp_secondary_reg_8-3 (high)

The following table shows the sys_cache_grp_secondary_reg_8-3 higher register bit assignments.

Table 3-585 por_rnsam_sys_cache_grp_secondary_reg_8-3 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	region#{index}_scndry_size	<p>Secondary memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	5'b00000
55:52	Reserved	Reserved	RO	-
51:32	region#{index}_scndry_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_HTG_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

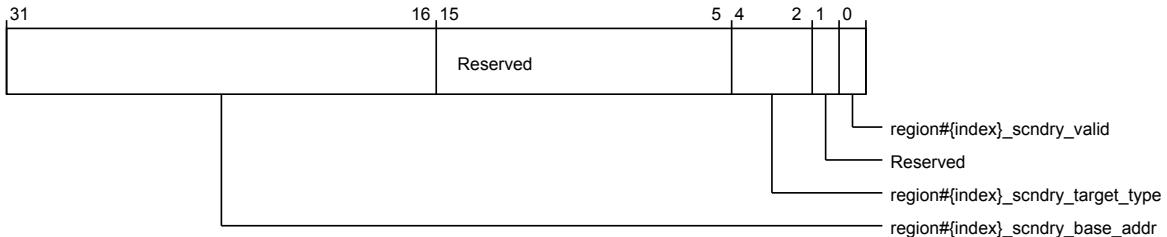


Figure 3-566 por_rnsam_sys_cache_grp_secondary_reg_8-3 (low)

The following table shows the sys_cache_grp_secondary_reg_8-3 lower register bit assignments.

Table 3-586 por_rnsam_sys_cache_grp_secondary_reg_8-3 (low)

Bits	Field name	Description	Type	Reset
31:16	region#{index}_scndry_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_HTG_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-
4:2	region#{index}_scndry_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000

Table 3-586 por_rnsam_sys_cache_grp_secondary_reg_8-3 (low) (continued)

Bits	Field name	Description	Type	Reset
1	Reserved	Reserved	RO	-
0	region#{index}_scndry_valid	<p>Secondary memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

hashed_target_grp_secondary_cfg1_reg_8-31

This register repeats 23 times. It parametrized by the index from 8 to 31. Configures secondary hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

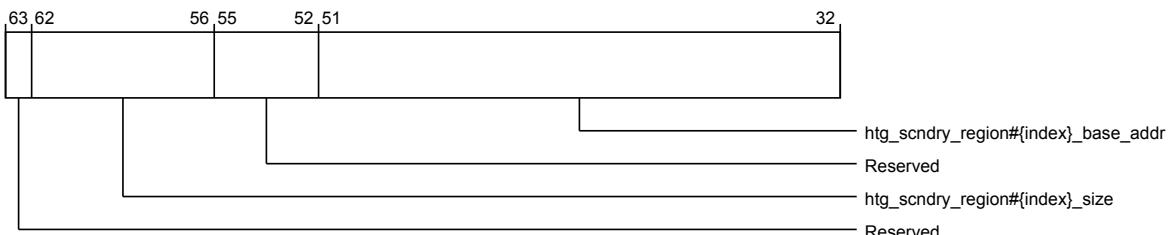


Figure 3-567 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (high)

The following table shows the hashed_target_grp_secondary_cfg1_reg_8-31 higher register bit assignments.

Table 3-587 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	htg_scndry_region#{index}_size	<p>Secondary memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	5'b00000
55:52	Reserved	Reserved	RO	-
51:32	htg_scndry_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

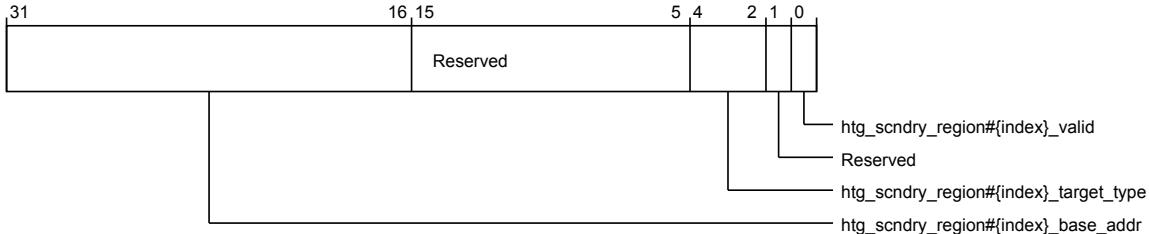


Figure 3-568 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (low)

The following table shows the hashed_target_grp_secondary_cfg1_reg_8-31 lower register bit assignments.

Table 3-588 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (low)

Bits	Field name	Description	Type	Reset
31:16	htg_scndry_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-
4:2	htg_scndry_region#{index}_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000
1	Reserved	Reserved	RO	-
0	htg_scndry_region#{index}_valid	<p>Secondary memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

hashed_target_grp_secondary_cfg1_reg_8-31

This register repeats 23 times. It parametrized by the index from 8 to 31. Configures secondary hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hE40
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

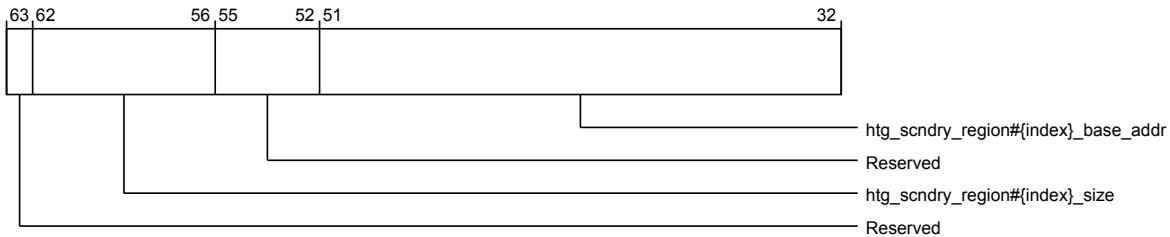


Figure 3-569 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (high)

The following table shows the hashed_target_grp_secondary_cfg1_reg_8-31 higher register bit assignments.

Table 3-589 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	htg_scndry_region#{index}_size	<p>Secondary memory region #{index} size</p> <p>CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).</p>	RW	5'b00000
55:52	Reserved	Reserved	RO	-
51:32	htg_scndry_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0

The following image shows the lower register bit assignments.

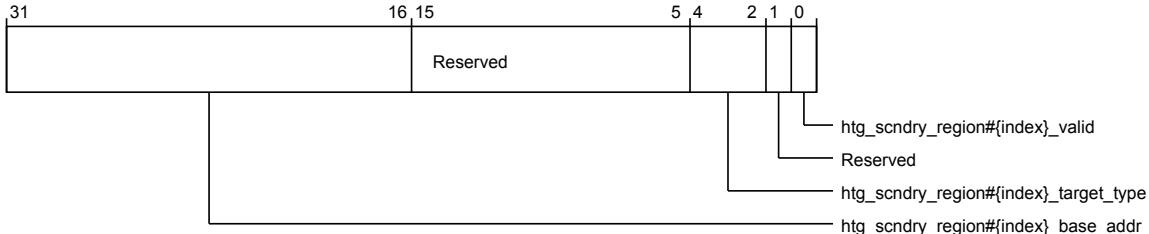


Figure 3-570 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (low)

The following table shows the hashed_target_grp_secondary_cfg1_reg_8-31 lower register bit assignments.

Table 3-590 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (low)

Bits	Field name	Description	Type	Reset
31:16	htg_scndry_region#{index}_base_addr	<p>Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM</p>	RW	36'h0
15:5	Reserved	Reserved	RO	-

Table 3-590 por_rnsam_hashed_target_grp_secondary_cfg1_reg_8-31 (low) (continued)

Bits	Field name	Description	Type	Reset
4:2	htg_scndry_region#{index}_target_type	<p>Indicates node type</p> <p>3'b000: HN-F</p> <p>3'b001: HN-I</p> <p>3'b010: CXRA</p> <p>3'b011: HN-P</p> <p>3'b100: PCI-CXRA</p> <p>Others: Reserved</p> <p>CONSTRAINT: Only applicable for RN-I</p>	RW	3'b000
1	Reserved	Reserved	RO	-
0	htg_scndry_region#{index}_valid	<p>Secondary memory region #{index} valid</p> <p>1'b0: Not valid</p> <p>1'b1: Valid for memory region comparison</p>	RW	1'b0

hashed_target_grp_secondary_cfg2_reg_0-31

This register repeats 31 times. It parametrized by the index from 0 to 31. Configures hashed memory regions

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3300 + # {8 * [0, 1, 2, .., 30, 31]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

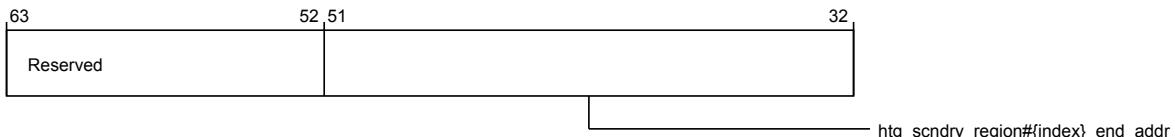


Figure 3-571 por_rnsam_hashed_target_grp_secondary_cfg2_reg_0-31 (high)

The following table shows the hashed_target_grp_secondary_cfg2_reg_0-31 higher register bit assignments.

Table 3-591 por_rnsam_hashed_target_grp_secondary_cfg2_reg_0-31 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	htg_scndry_region#{index}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAMHTG_RCOMP_LSB_PARAM	RW	36'b00000000000000000000000000000000

The following image shows the lower register bit assignments.

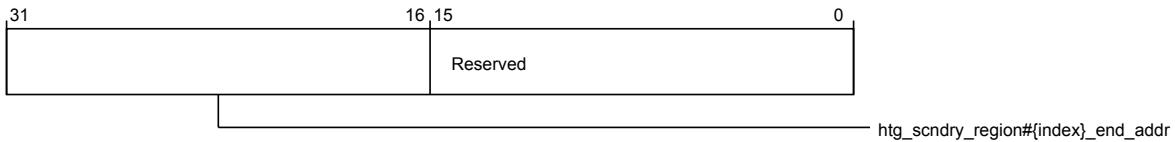


Figure 3-572 por_rnsam_hashed_target_grp_secondary_cfg2_reg_0-31 (low)

The following table shows the hashed_target_grp_secondary_cfg2_reg_0-31 lower register bit assignments.

Table 3-592 por_rnsam_hashed_target_grp_secondary_cfg2_reg_0-31 (low)

Bits	Field name	Description	Type	Reset
31:16	htg_scndry_region#{index}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_RNSAM_HTG_RCOMP_LSB_PARAM	RW	36'b00000000000000000000000000000000
15:0	Reserved	Reserved	RO	-

hashed_target_grp_hash_cntl_reg_0-31

This register repeats 31 times. It parametrized by the index from 0 to 31. Configures HTG hash type
Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3400 + # {8 * [0, 1, 2, .., 30, 31]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

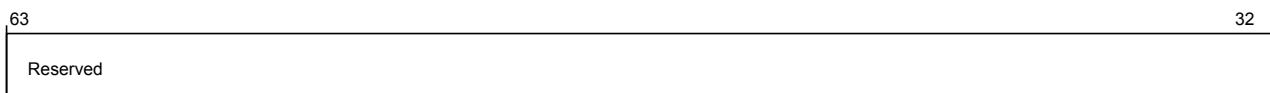


Figure 3-573 por_rnsam_hashed_target_grp_hash_cntl_reg_0-31 (high)

The following table shows the hashed_target_grp_hash_cntl_reg_0-31 higher register bit assignments.

Table 3-593 por_rnsam_hashed_target_grp_hash_cntl_reg_0-31 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

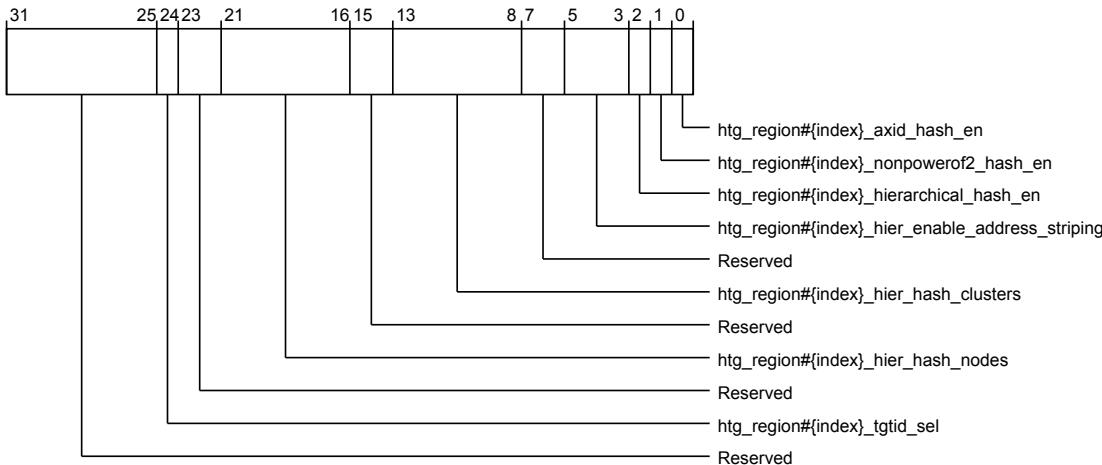


Figure 3-574 por_rnsam_hashed_target_grp_hash_cntl_reg_0-31 (low)

The following table shows the hashed_target_grp_hash_cntl_reg_0-31 lower register bit assignments.

Table 3-594 por_rnsam_hashed_target_grp_hash_cntl_reg_0-31 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	htg_region#{index}_tgtid_sel	Select the TgtID's from HNF or HNP trgt tables 1'b0: Default, selects from HNF table 1'b1: selects from HNP table	RW	1'b0
23:22	Reserved	Reserved	RO	-
21:16	htg_region#{index}_hier_hash_nodes	Hierarchical hashing mode, define number of nodes in each cluster	RW	6'h0
15:14	Reserved	Reserved	RO	-
13:8	htg_region#{index}_hier_hash_clusters	Hierarchical hashing mode, define number of clusters groups	RW	6'h0
7:6	Reserved	Reserved	RO	-
5:3	htg_region#{index}_hier_enable_address_striping	Hierarchical hashing: configure address striping at second hierarchy hash 3'b000: no address striping 3'b001: stripe address bit [6] 3'b010: stripe address bit [7:6] 3'b011: stripe address bit [8:6] 3'b100: stripe address bit [9:6] 3'b101: stripe address bit [10:6] 3'b110: stripe address bit [11:6] others: Reserved	RW	3'b0

Table 3-594 por_rnsam_hashed_target_grp_hash_cntl_reg_0-31 (low) (continued)

Bits	Field name	Description	Type	Reset
2	htg_region#{index}_hierarchical_hash_en	Hierarchical Hashing mode enable configure bit	RW	1'b0
1	htg_region#{index}_nonpowerof2_hash_en	Non power of two Hashing mode enable cconfigure bit	RW	1'b0
0	htg_region#{index}_axid_hash_en	AXID based Hashing mode enable configure bit	RW	1'b0

sys_cache_group_hn_count

Indicates number of HN-F/HN-P's in hashed target groups 0 to 7.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hEA0

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

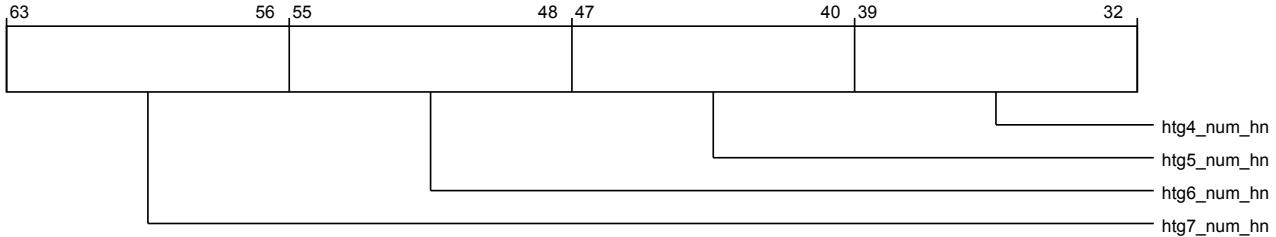


Figure 3-575 por_rnsam_sys_cache_group_hn_count (high)

The following table shows the sys_cache_group_hn_count higher register bit assignments.

Table 3-595 por_rnsam_sys_cache_group_hn_count (high)

Bits	Field name	Description	Type	Reset
63:56	htg7_num_hn	HN count for hashed target group 7	RW	8'h00
55:48	htg6_num_hn	HN count for hashed target group 6	RW	8'h00
47:40	htg5_num_hn	HN count for hashed target group 5	RW	8'h00
39:32	htg4_num_hn	HN count for hashed target group 4	RW	8'h00

The following image shows the lower register bit assignments.

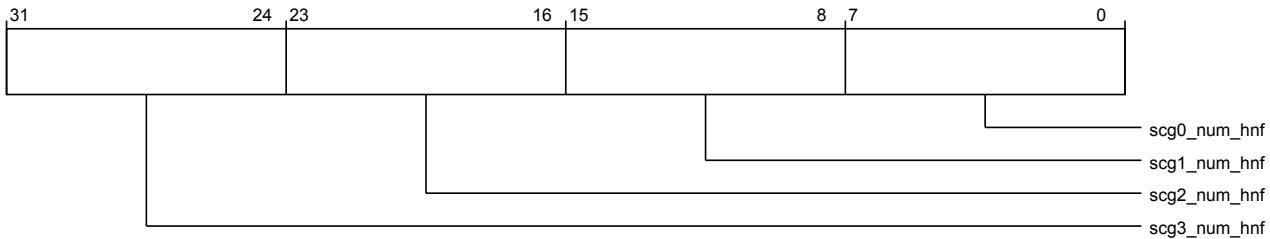


Figure 3-576 por_rnsam_sys_cache_group_hn_count (low)

The following table shows the sys_cache_group_hn_count lower register bit assignments.

Table 3-596 por_rnsam_sys_cache_group_hn_count (low)

Bits	Field name	Description	Type	Reset
31:24	scg3_num_hnf	HN count for hashed target group 3	RW	8'h00
23:16	scg2_num_hnf	HN count for hashed target group 2	RW	8'h00
15:8	scg1_num_hnf	HN count for hashed target group 1	RW	8'h00
7:0	scg0_num_hnf	HN count for hashed target group 0	RW	8'h00

hashed_target_group_hn_count_reg_2-3

This register repeats 1 times. It parametrized by the index from 2 to 3. Indicates number of HN-F/HN-P's in hashed target groups #{{index}*8} to #{{index}*8 + 7}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hEA0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

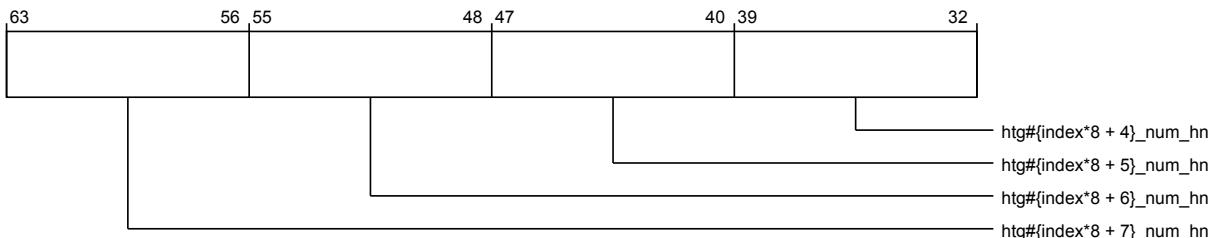


Figure 3-577 por_rnsam_hashed_target_group_hn_count_reg_2-3 (high)

The following table shows the hashed_target_group_hn_count_reg_2-3 higher register bit assignments.

Table 3-597 por_rnsam_hashed_target_group_hn_count_reg_2-3 (high)

Bits	Field name	Description	Type	Reset
63:56	htg#{index*8 + 7}_num_hn	<p>HN count for hashed target group 7</p>	RW	8'h00
55:48	htg#{index*8 + 6}_num_hn	<p>HN count for hashed target group 6</p>	RW	8'h00
47:40	htg#{index*8 + 5}_num_hn	<p>HN count for hashed target group 5</p>	RW	8'h00
39:32	htg#{index*8 + 4}_num_hn	<p>HN count for hashed target group 4</p>	RW	8'h00

The following image shows the lower register bit assignments.

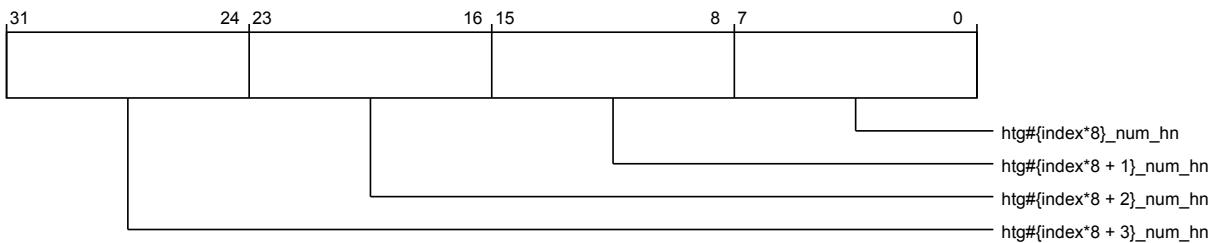


Figure 3-578 por_rnsam_hashed_target_group_hn_count_reg_2-3 (low)

The following table shows the hashed_target_group_hn_count_reg_2-3 lower register bit assignments.

Table 3-598 por_rnsam_hashed_target_group_hn_count_reg_2-3 (low)

Bits	Field name	Description	Type	Reset
31:24	htg#{index*8 + 3}_num_hn	<p>HN count for hashed target group 3</p>	RW	8'h00
23:16	htg#{index*8 + 2}_num_hn	<p>HN count for hashed target group 2</p>	RW	8'h00
15:8	htg#{index*8 + 1}_num_hn	<p>HN count for hashed target group 1</p>	RW	8'h00
7:0	htg#{index*8}_num_hn	<p>HN count for hashed target group 0</p>	RW	8'h00

hashed_target_group_hn_count_reg_2-3

This register repeats 1 times. It parametrized by the index from 2 to 3. Indicates number of HN-F/HN-P's in hashed target groups #{{index}*8} to #{{index}*8 + 7}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hEA0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

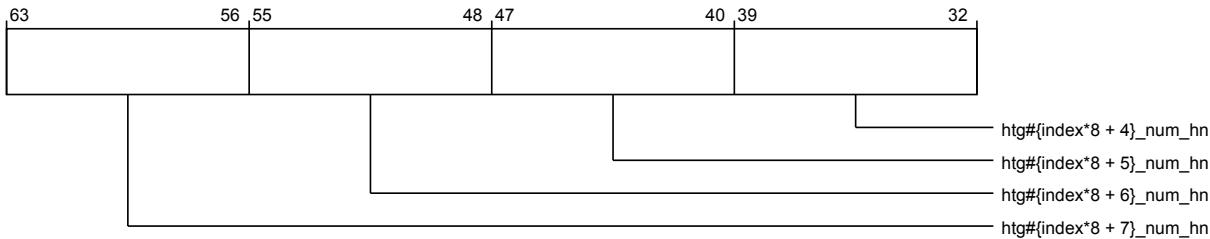


Figure 3-579 por_rnsam_hashed_target_group_hn_count_reg_2-3 (high)

The following table shows the hashed_target_group_hn_count_reg_2-3 higher register bit assignments.

Table 3-599 por_rnsam_hashed_target_group_hn_count_reg_2-3 (high)

Bits	Field name	Description	Type	Reset
63:56	htg#{index*8 + 7}_num_hn	<p>HN count for hashed target group 7</p>	RW	8'h00
55:48	htg#{index*8 + 6}_num_hn	<p>HN count for hashed target group 6</p>	RW	8'h00
47:40	htg#{index*8 + 5}_num_hn	<p>HN count for hashed target group 5</p>	RW	8'h00
39:32	htg#{index*8 + 4}_num_hn	<p>HN count for hashed target group 4</p>	RW	8'h00

The following image shows the lower register bit assignments.

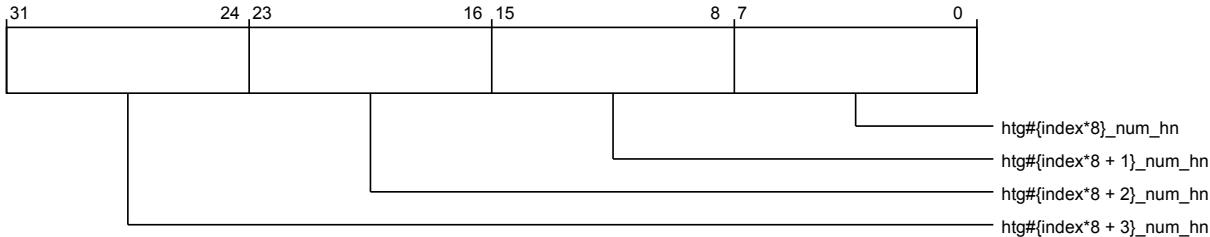


Figure 3-580 por_rnsam_hashed_target_group_hn_count_reg_2-3 (low)

The following table shows the hashed_target_group_hn_count_reg_2-3 lower register bit assignments.

Table 3-600 por_rnsam_hashed_target_group_hn_count_reg_2-3 (low)

Bits	Field name	Description	Type	Reset
31:24	htg#{index*8 + 3}_num_hn	<p>HN count for hashed target group 3</p>	RW	8'h00
23:16	htg#{index*8 + 2}_num_hn	<p>HN count for hashed target group 2</p>	RW	8'h00
15:8	htg#{index*8 + 1}_num_hn	<p>HN count for hashed target group 1</p>	RW	8'h00
7:0	htg#{index*8}_num_hn	<p>HN count for hashed target group 0</p>	RW	8'h00

sys_cache_grp_nonhash_nodeid

Configures non-hashed node IDs for hashed target groups 1 to 5. NOTE: Only applicable in the non-hashed mode.

Its characteristics are:

Type	RW

Register width (Bits)	64
Address offset	16'hEC0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

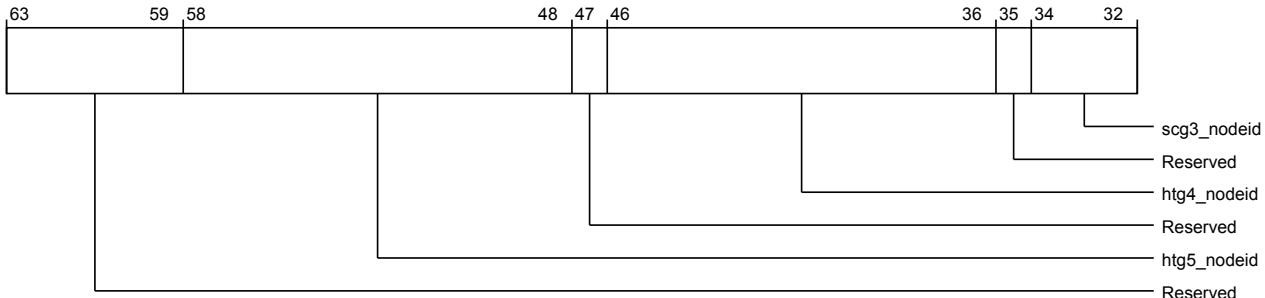


Figure 3-581 por_rnsam_sys_cache_grp_nonhash_nodeid (high)

The following table shows the sys cache grp nonhash nodeid higher register bit assignments.

Table 3-601 por_rnsam_sys_cache_grp_nonhash_nodeid (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	htg5_nodeid	Non-hashed node ID for Hashed target group 5	RW	11'b000000000000
47	Reserved	Reserved	RO	-
46:36	htg4_nodeid	Non-hashed node ID for Hashed target group 4	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	scg3_nodeid	Non-hashed node ID for Hashed target group 3	RW	11'b000000000000

The following image shows the lower register bit assignments.

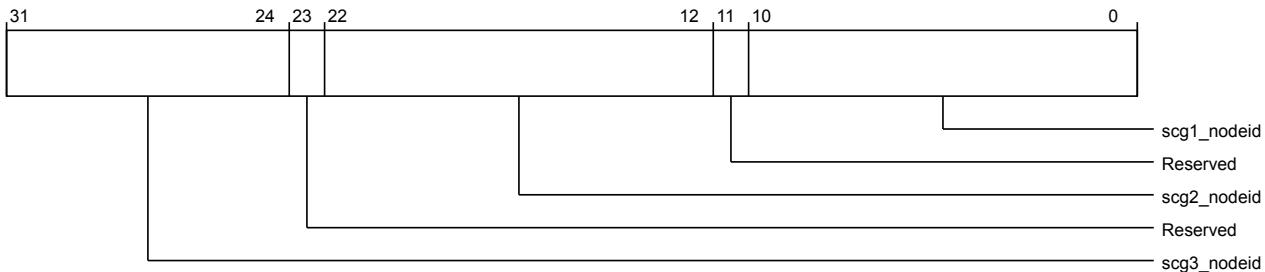


Figure 3-582 por_rnsam_sys_cache_grp_nonhash_nodeid (low)

The following table shows the sys_cache_grp_nonhash_nodeid lower register bit assignments.

Table 3-602 por_rnsam_sys_cache_grp_nonhash_nodeid (low)

Bits	Field name	Description	Type	Reset
31:24	scg3_nodeid	Non-hashed node ID for Hashed target group 3	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	scg2_nodeid	Non-hashed node ID for Hashed target group 2	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	scg1_nodeid	Non-hashed node ID for Hashed target group 1	RW	11'b000000000000

hashed_target_grp_nonhash_nodeid_reg_6-6

This register repeats 0 times. It parametrized by the index from 6 to 6. Configures non-hashed node IDs for hashed target groups # $\{index*5 + 1\}$ to # $\{index*5 + 5\}$. NOTE: Only applicable in the non-hashed mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hEC0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

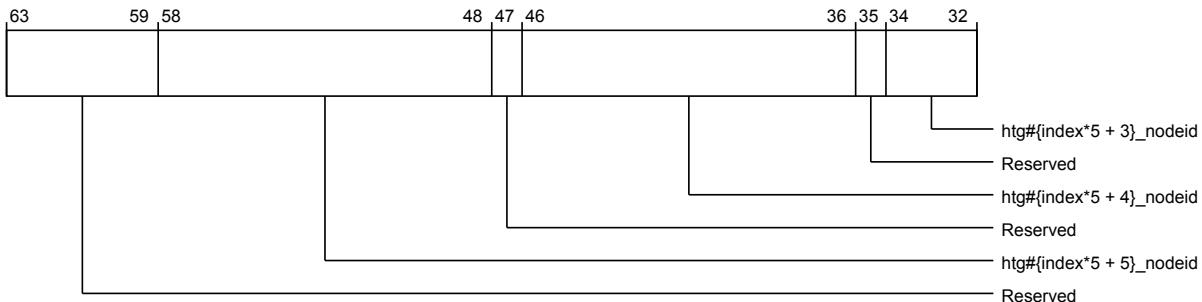


Figure 3-583 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (high)

The following table shows the hashed_target_grp_nonhash_nodeid_reg_6-6 higher register bit assignments.

Table 3-603 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	htg#{index*5 + 5}_nodeid	<p>Non-hashed node ID for Hashed target group # $\{index*5 + 5\}$ </p>	RW	11'b000000000000
47	Reserved	Reserved	RO	-

Table 3-603 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (high) (continued)

Bits	Field name	Description	Type	Reset
46:36	htg#{index*5 + 4}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 4}</p>	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	htg#{index*5 + 3}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 3}</p>	RW	11'b000000000000

The following image shows the lower register bit assignments.

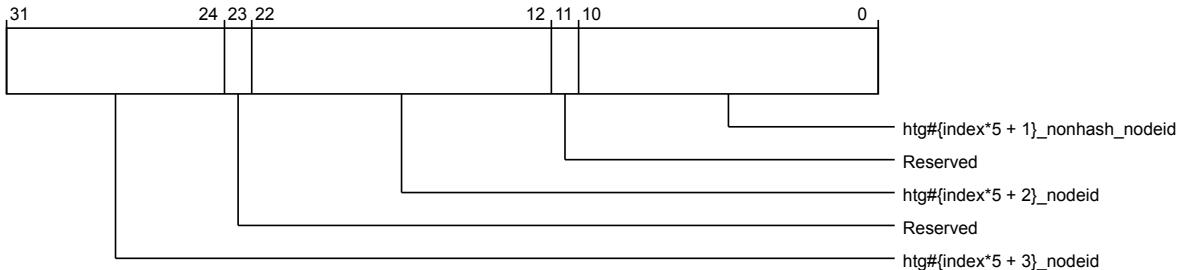


Figure 3-584 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (low)

The following table shows the hashed_target_grp_nonhash_nodeid_reg_6-6 lower register bit assignments.

Table 3-604 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (low)

Bits	Field name	Description	Type	Reset
31:24	htg#{index*5 + 3}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 3}</p>	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	htg#{index*5 + 2}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 2}</p>	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	htg#{index*5 + 1}_nonhash_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 1}</p>	RW	11'b000000000000

hashed_target_grp_nonhash_nodeid_reg_6-6

This register repeats 0 times. It parametrized by the index from 6 to 6. Configures non-hashed node IDs for hashed target groups #{index*5 + 1} to #{index*5 + 5}. NOTE: Only applicable in the non-hashed mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hEC0
Register reset	64'b0

Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
--------------------------	--

The following image shows the higher register bit assignments.

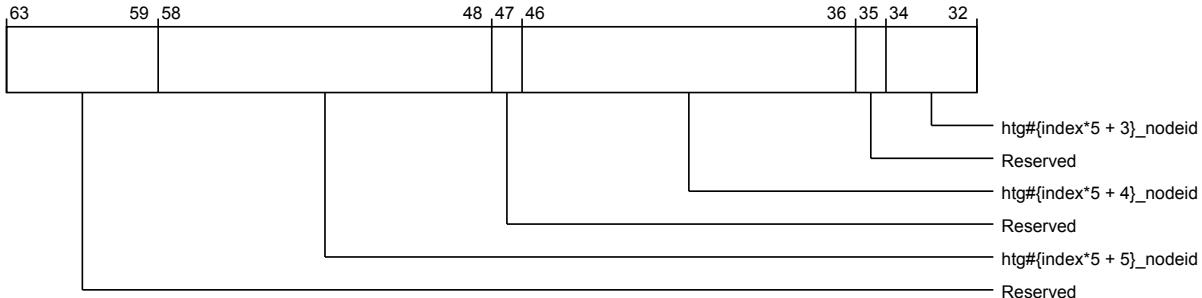


Figure 3-585 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (high)

The following table shows the hashed_target_grp_nonhash_nodeid_reg_6-6 higher register bit assignments.

Table 3-605 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	htg#{index*5 + 5}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 5}</p>	RW	11'b000000000000
47	Reserved	Reserved	RO	-
46:36	htg#{index*5 + 4}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 4}</p>	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	htg#{index*5 + 3}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 3}</p>	RW	11'b000000000000

The following image shows the lower register bit assignments.

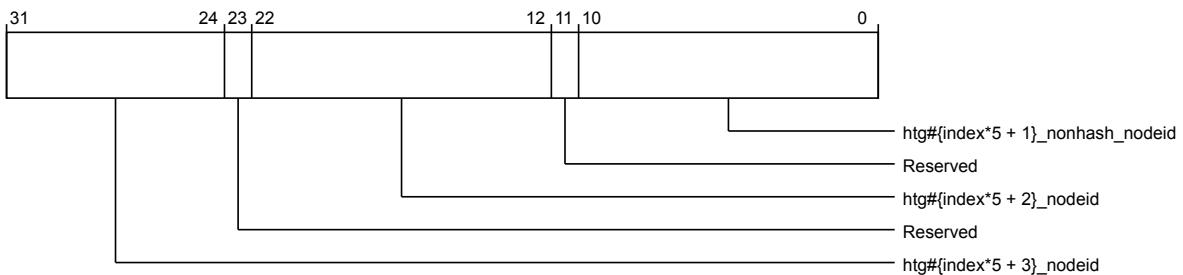


Figure 3-586 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (low)

The following table shows the hashed_target_grp_nonhash_nodeid_reg_6-6 lower register bit assignments.

Table 3-606 por_rnsam_hashed_target_grp_nonhash_nodeid_reg_6-6 (low)

Bits	Field name	Description	Type	Reset
31:24	htg#{index*5 + 3}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 3}</p>	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	htg#{index*5 + 2}_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 2}</p>	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	htg#{index*5 + 1}_nonhash_nodeid	<p>Non-hashed node ID for Hashed target group #{index*5 + 1}</p>	RW	11'b000000000000

sys_cache_grp_hn_nodeid_reg_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Configures HNF node IDs for hashed target groups. Controls target HNF node IDs #{index*4} to #{index*4 + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF00 + #{8 * [0, 1, 2, .., 14, 15]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

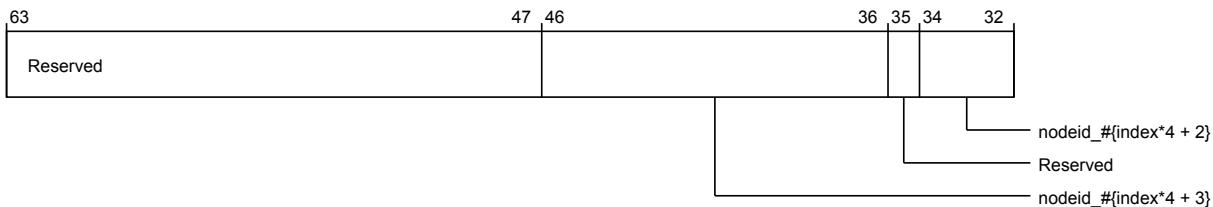


Figure 3-587 por_rnsam_sys_cache_grp_hn_nodeid_reg_0-15 (high)

The following table shows the sys_cache_grp_hn_nodeid_reg_0-15 higher register bit assignments.

Table 3-607 por_rnsam_sys_cache_grp_hn_nodeid_reg_0-15 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	nodeid_#{index*4 + 3}	HNF target node ID #{index*4 + 3}	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	nodeid_#{index*4 + 2}	HNF target node ID #{index*4 + 2}	RW	11'b000000000000

The following image shows the lower register bit assignments.

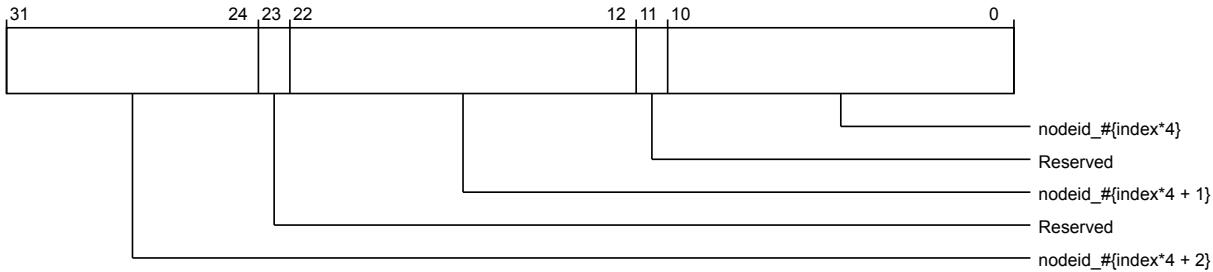


Figure 3-588 por_rnsam_sys_cache_grp_hn_nodeid_reg_0-15 (low)

The following table shows the sys_cache_grp_hn_nodeid_reg_0-15 lower register bit assignments.

Table 3-608 por_rnsam_sys_cache_grp_hn_nodeid_reg_0-15 (low)

Bits	Field name	Description	Type	Reset
31:24	nodeid_{index*4 + 2}	HNF target node ID #{index*4 + 2}	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	nodeid_{index*4 + 1}	HNF target node ID #{index*4 + 1}	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	nodeid_{index*4}	HNF target node ID #{index*4}	RW	11'b000000000000

hashed_target_grp_hnf_nodeid_reg_32-31

This register repeats -1 times. It parametrized by the index from 32 to 31. Configures HNF node IDs for hashed target groups. Controls target HNF node IDs #{index*4} to #{index*4 + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

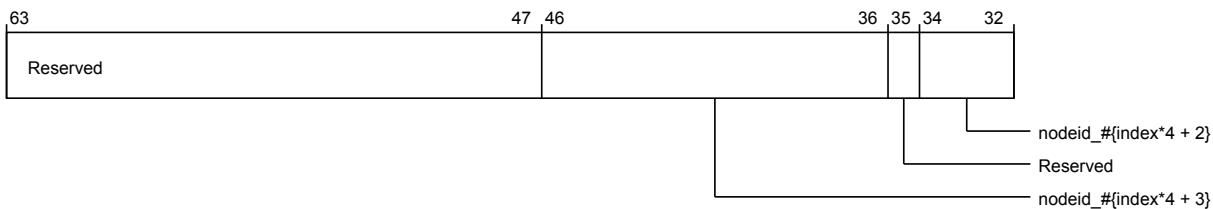


Figure 3-589 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (high)

The following table shows the hashed_target_grp_hnf_nodeid_reg_32-31 higher register bit assignments.

Table 3-609 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	nodeid_{index*4 + 3}	<p>HNF target node ID #{index*4 + 3}</p>	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	nodeid_{index*4 + 2}	<p>HNF target node ID #{index*4 + 2}</p>	RW	11'b000000000000

The following image shows the lower register bit assignments.

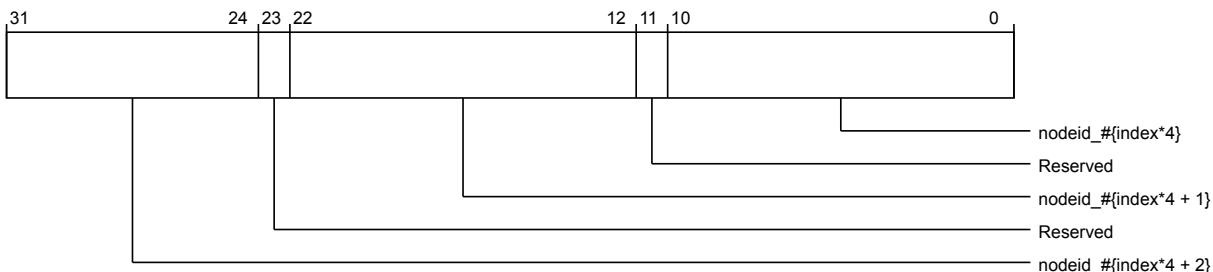


Figure 3-590 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (low)

The following table shows the hashed_target_grp_hnf_nodeid_reg_32-31 lower register bit assignments.

Table 3-610 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (low)

Bits	Field name	Description	Type	Reset
31:24	nodeid_{index*4 + 2}	<p>HNF target node ID #{index*4 + 2}</p>	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	nodeid_{index*4 + 1}	<p>HNF target node ID #{index*4 + 1}</p>	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	nodeid_{index*4}	<p>HNF target node ID #{index*4}</p>	RW	11'b000000000000

hashed_target_grp_hnf_nodeid_reg_32-31

This register repeats -1 times. It parametrized by the index from 32 to 31. Configures HNF node IDs for hashed target groups. Controls target HNF node IDs #{index*4} to #{index*4 + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

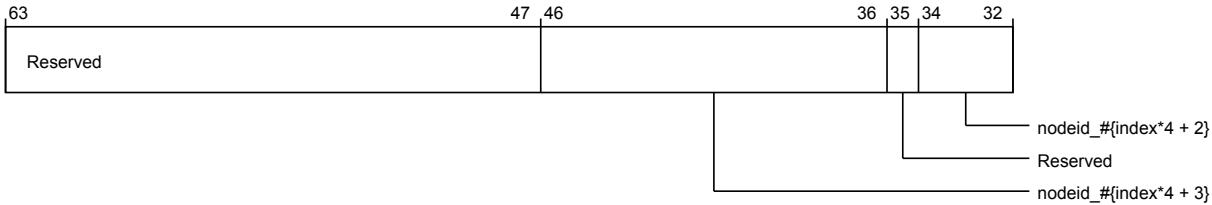


Figure 3-591 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (high)

The following table shows the hashed_target_grp_hnf_nodeid_reg_32-31 higher register bit assignments.

Table 3-611 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	nodeid_{index*4 + 3}	<p>HNF target node ID #{index*4 + 3}</p>	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	nodeid_{index*4 + 2}	<p>HNF target node ID #{index*4 + 2}</p>	RW	11'b000000000000

The following image shows the lower register bit assignments.

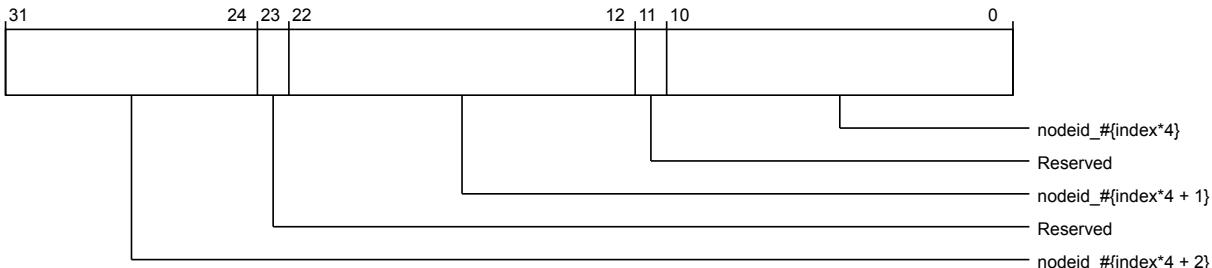


Figure 3-592 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (low)

The following table shows the hashed_target_grp_hnf_nodeid_reg_32-31 lower register bit assignments.

Table 3-612 por_rnsam_hashed_target_grp_hnf_nodeid_reg_32-31 (low)

Bits	Field name	Description	Type	Reset
31:24	nodeid_{index*4 + 2}	<p>HNF target node ID #{index*4 + 2}</p>	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	nodeid_{index*4 + 1}	<p>HNF target node ID #{index*4 + 1}</p>	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	nodeid_{index*4}	<p>HNF target node ID #{index*4}</p>	RW	11'b000000000000

hashed_target_grp_hnp_nodeid_reg_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Configures HNP node IDs for hashed target groups. Controls target HNP node IDs # {index*4} to # {index*4 + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3600 + #{8 * [0, 1, 2, .., 14, 15]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

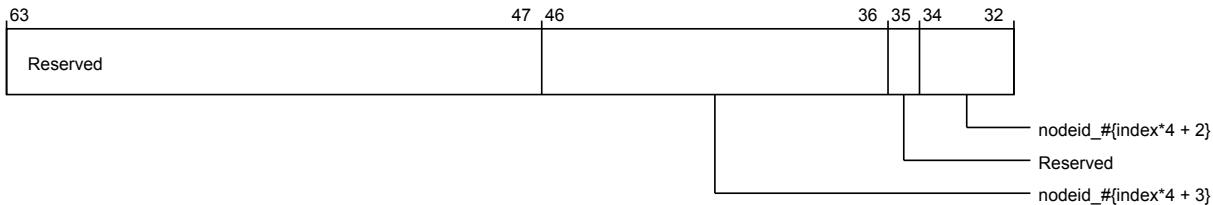


Figure 3-593 por_rnsam_hashed_target_grp_hnp_nodeid_reg_0-15 (high)

The following table shows the hashed_target_grp_hnp_nodeid_reg_0-15 higher register bit assignments.

Table 3-613 por_rnsam_hashed_target_grp_hnp_nodeid_reg_0-15 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	nodeid_#{index*4 + 3}	HNP target node ID #{index*4 + 3}	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	nodeid_#{index*4 + 2}	HNP target node ID #{index*4 + 2}	RW	11'b000000000000

The following image shows the lower register bit assignments.

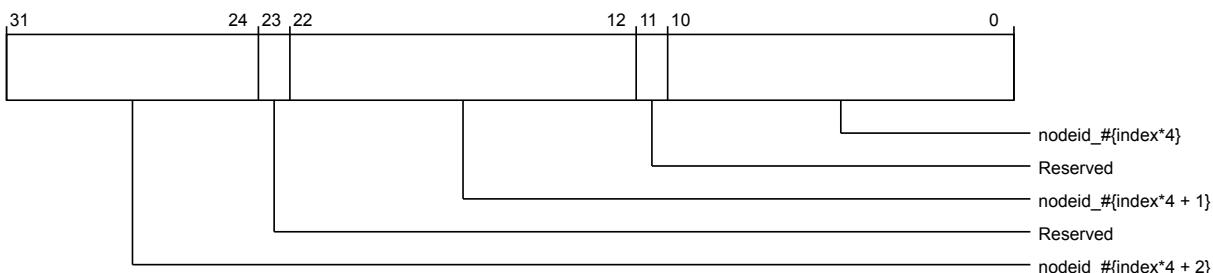


Figure 3-594 por_rnsam_hashed_target_grp_hnp_nodeid_reg_0-15 (low)

The following table shows the hashed_target_grp_hnp_nodeid_reg_0-15 lower register bit assignments.

Table 3-614 por_rnsam_hashed_target_grp_hnp_nodeid_reg_0-15 (low)

Bits	Field name	Description	Type	Reset
31:24	nodeid_{index*4 + 2}	HNP target node ID #{index*4 + 2}	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	nodeid_{index*4 + 1}	HNP target node ID #{index*4 + 1}	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	nodeid_{index*4}	HNP target node ID #{index*4}	RW	11'b000000000000

sys_cache_grp_cal_mode_reg

Configures the HN CAL mode support for all hashed target groups.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1

Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.

Secure group override por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments

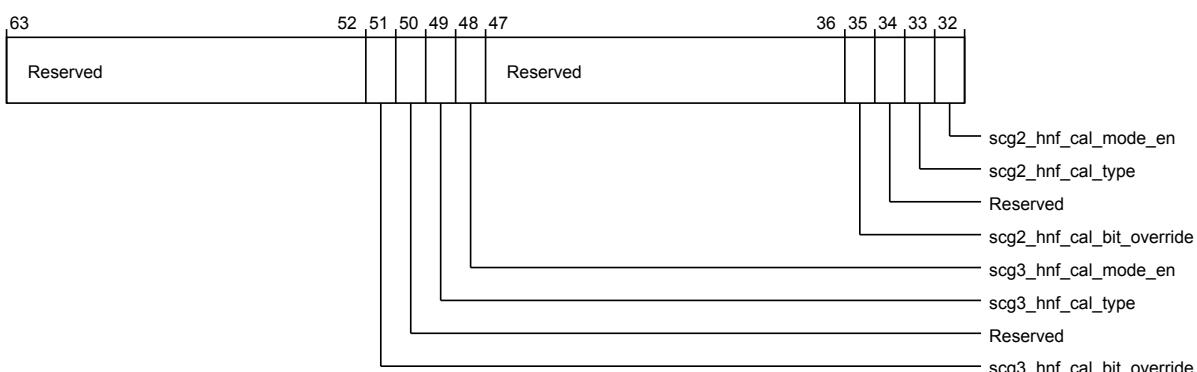


Figure 3-595 por_rnsam_sys_cache_grp_cal_mode_reg (high)

The following table shows the sys cache grp cal mode reg higher register bit assignments.

Table 3-615 por_rnsam_sys_cache_grp_cal_mode_reg (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51	scg3_hnf_cal_bit_override	Configuration to choose LSB/MSB bit to override Device ID for HTG 3 1'b0: Hash MSB bit to override Device ID 1'b1: Hash LSB bit to override Device ID	RW	1'b0
50	Reserved	Reserved	RO	-
49	scg3_hnf_cal_type	Enables type of HN CAL for HTG 3 1'b0: CAL2 mode 1'b1: CAL4 mode	RW	1'b0
48	scg3_hnf_cal_mode_en	Enables support for HN CAL for HTG 3	RW	1'b0
47:36	Reserved	Reserved	RO	-
35	scg2_hnf_cal_bit_override	Configuration to choose LSB/MSB bit to override Device ID for HTG 2 1'b0: Hash MSB bit to override Device ID 1'b1: Hash LSB bit to override Device ID	RW	1'b0
34	Reserved	Reserved	RO	-
33	scg2_hnf_cal_type	Enables type of HN CAL for HTG 2 1'b0: CAL2 mode 1'b1: CAL4 mode	RW	1'b0
32	scg2_hnf_cal_mode_en	Enables support for HN CAL for HTG 2	RW	1'b0

The following image shows the lower register bit assignments.

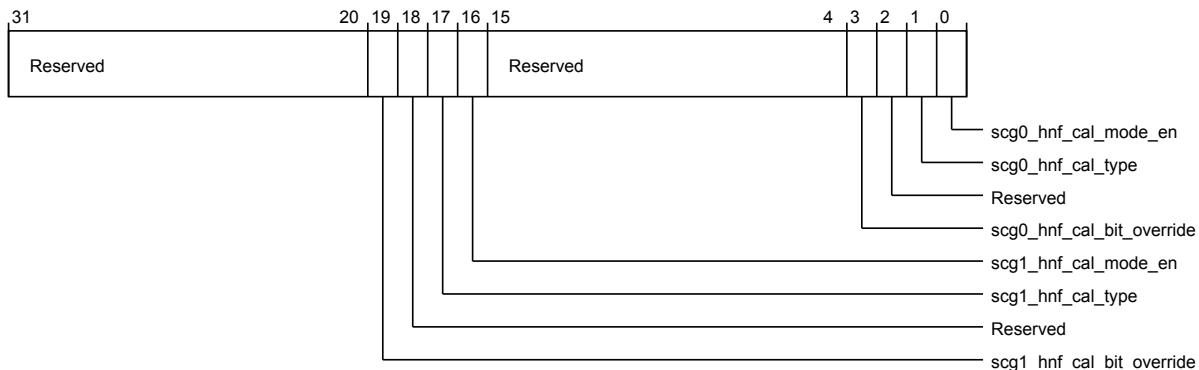


Figure 3-596 por_rnsam_sys_cache_grp_cal_mode_reg (low)

The following table shows the sys_cache_grp_cal_mode_reg lower register bit assignments.

Table 3-616 por_rnsam_sys_cache_grp_cal_mode_reg (low)

Bits	Field name	Description	Type	Reset
31:20	Reserved	Reserved	RO	-
19	scg1_hnf_cal_bit_override	Configuration to choose LSB/MSB bit to override Device ID for HTG 1 1'b0: Hash MSB bit to override Device ID 1'b1: Hash LSB bit to override Device ID	RW	1'b0
18	Reserved	Reserved	RO	-
17	scg1_hnf_cal_type	Enables type of HN CAL for HTG 1 1'b0: CAL2 mode 1'b1: CAL4 mode	RW	1'b0
16	scg1_hnf_cal_mode_en	Enables support for HN CAL for HTG 1	RW	1'b0
15:4	Reserved	Reserved	RO	-
3	scg0_hnf_cal_bit_override	Configuration to choose LSB/MSB bit to override Device ID for HTG 0 1'b0: Hash MSB bit to override Device ID 1'b1: Hash LSB bit to override Device ID	RW	1'b0
2	Reserved	Reserved	RO	-
1	scg0_hnf_cal_type	Enables type of HN CAL for HTG 0 1'b0: CAL2 mode 1'b1: CAL4 mode	RW	1'b0
0	scg0_hnf_cal_mode_en	Enables support for HN CAL for HTG 0	RW	1'b0

hashed_target_grp_cal_mode_reg_4-7

This register repeats 3 times. It parametrized by the index from 4 to 7. Configures the HN CAL mode support for all hashed target groups.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1120
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

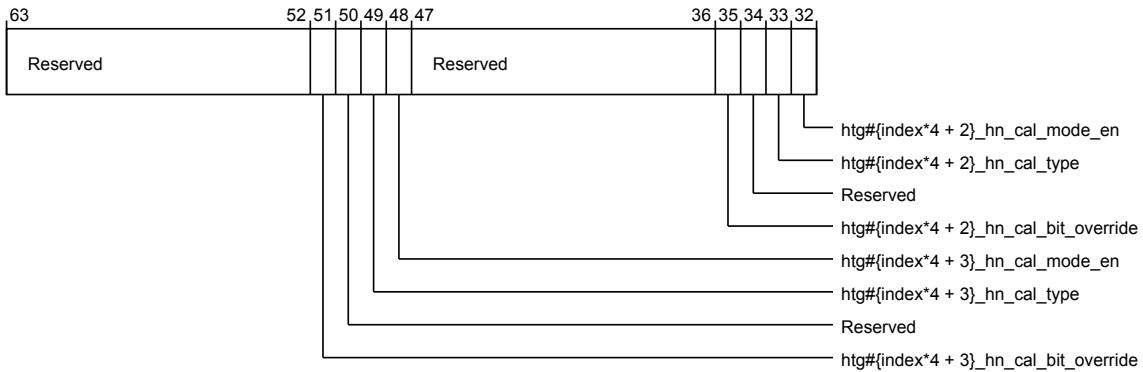


Figure 3-597 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (high)

The following table shows the hashed_target_grp_cal_mode_reg_4-7 higher register bit assignments.

Table 3-617 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51	htg#{index*4 + 3}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4 + 3}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
50	Reserved	Reserved	RO	-
49	htg#{index*4 + 3}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4 + 3}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
48	htg#{index*4 + 3}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4 + 3}</p>	RW	1'b0
47:36	Reserved	Reserved	RO	-
35	htg#{index*4 + 2}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4 + 2}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
34	Reserved	Reserved	RO	-
33	htg#{index*4 + 2}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4 + 2}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
32	htg#{index*4 + 2}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4 + 2}</p>	RW	1'b0

The following image shows the lower register bit assignments.

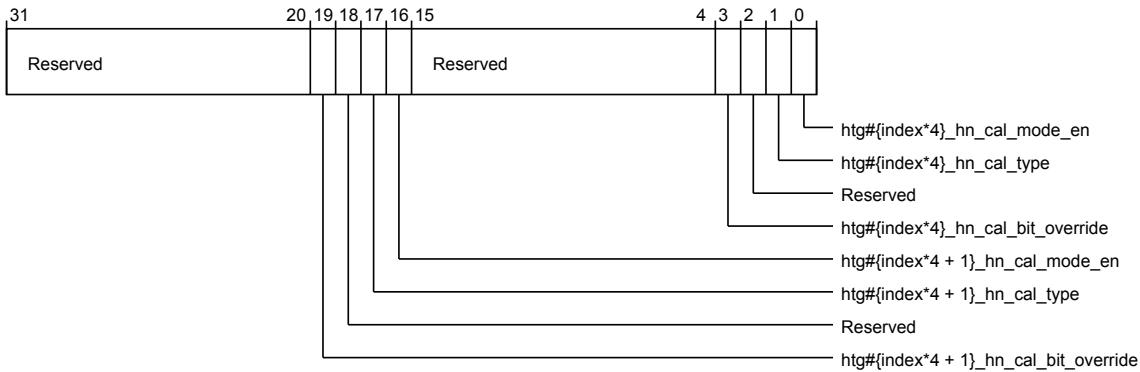


Figure 3-598 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (low)

The following table shows the hashed_target_grp_cal_mode_reg_4-7 lower register bit assignments.

Table 3-618 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (low)

Bits	Field name	Description	Type	Reset
31:20	Reserved	Reserved	RO	-
19	htg#{index*4 + 1}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4 + 1}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
18	Reserved	Reserved	RO	-
17	htg#{index*4 + 1}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4 + 1}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
16	htg#{index*4 + 1}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4 + 1}</p>	RW	1'b0
15:4	Reserved	Reserved	RO	-
3	htg#{index*4}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
2	Reserved	Reserved	RO	-
1	htg#{index*4}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
0	htg#{index*4}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4}</p>	RW	1'b0

hashed_target_grp_cal_mode_reg_4-7

This register repeats 3 times. It parametrized by the index from 4 to 7. Configures the HN CAL mode support for all hashed target groups.

Its characteristics are:

Type RW
Register width (Bits) 64

Address offset	16'h1120
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

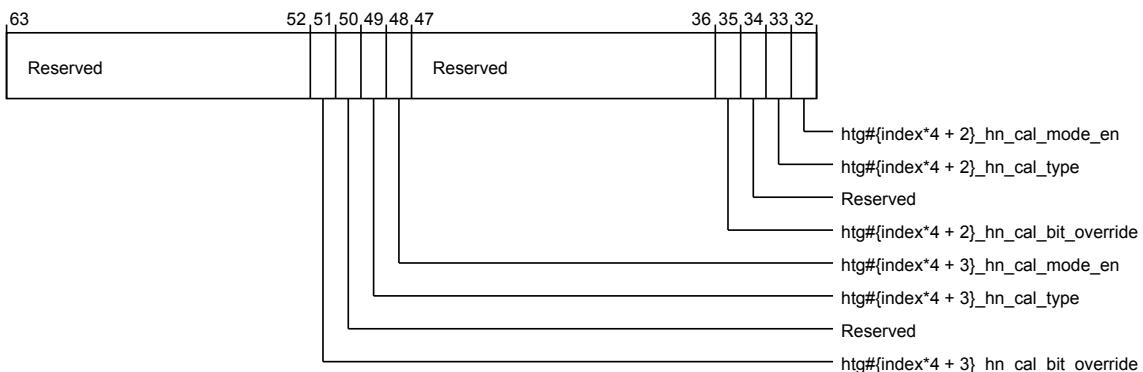


Figure 3-599 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (high)

The following table shows the hashed_target_grp_cal_mode_reg_4-7 higher register bit assignments.

Table 3-619 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51	htg#{index*4 + 3}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4 + 3}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
50	Reserved	Reserved	RO	-
49	htg#{index*4 + 3}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4 + 3}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
48	htg#{index*4 + 3}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4 + 3}</p>	RW	1'b0
47:36	Reserved	Reserved	RO	-
35	htg#{index*4 + 2}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4 + 2}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
34	Reserved	Reserved	RO	-

Table 3-619 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (high) (continued)

Bits	Field name	Description	Type	Reset
33	htg#{index*4 + 2}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4 + 2}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
32	htg#{index*4 + 2}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4 + 2}</p>	RW	1'b0

The following image shows the lower register bit assignments.

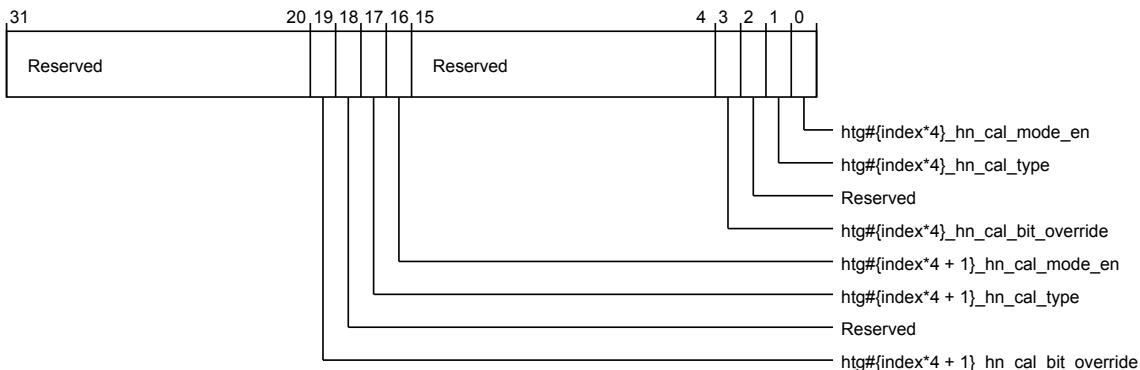


Figure 3-600 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (low)

The following table shows the hashed_target_grp_cal_mode_reg_4-7 lower register bit assignments.

Table 3-620 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (low)

Bits	Field name	Description	Type	Reset
31:20	Reserved	Reserved	RO	-
19	htg#{index*4 + 1}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4 + 1}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
18	Reserved	Reserved	RO	-
17	htg#{index*4 + 1}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4 + 1}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
16	htg#{index*4 + 1}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4 + 1}</p>	RW	1'b0
15:4	Reserved	Reserved	RO	-
3	htg#{index*4}_hn_cal_bit_override	<p>Configuration to choose LSB/MSB bit to override Device ID for HTG #{index*4}</p> <p>1'b0: Hash MSB bit to override Device ID</p> <p>1'b1: Hash LSB bit to override Device ID</p>	RW	1'b0
2	Reserved	Reserved	RO	-

Table 3-620 por_rnsam_hashed_target_grp_cal_mode_reg_4-7 (low) (continued)

Bits	Field name	Description	Type	Reset
1	htg#{index*4}_hn_cal_type	<p>Enables type of HN CAL for HTG #{index*4}</p> <p>1'b0: CAL2 mode</p> <p>1'b1: CAL4 mode</p>	RW	1'b0
0	htg#{index*4}_hn_cal_mode_en	<p>Enables support for HN CAL for HTG #{index*4}</p>	RW	1'b0

sys_cache_grp_hn_cpa_en_reg

Configures CCIX port aggregation mode for hashed HNF node IDs

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1180

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

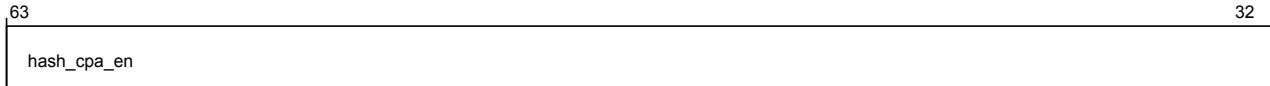


Figure 3-601 por_rnsam_sys_cache_grp_hn_cpa_en_reg (high)

The following table shows the sys_cache_grp_hn_cpa_en_reg higher register bit assignments.

Table 3-621 por_rnsam_sys_cache_grp_hn_cpa_en_reg (high)

Bits	Field name	Description	Type	Reset
63:32	hash_cpa_en	Enable CPA for each hashed HNF node ID	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

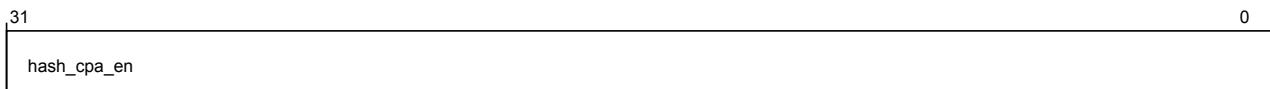


Figure 3-602 por_rnsam_sys_cache_grp_hn_cpa_en_reg (low)

The following table shows the sys_cache_grp_hn_cpa_en_reg lower register bit assignments.

Table 3-622 por_rnsam_sys_cache_grp_hn_cpa_en_reg (low)

Bits	Field name	Description	Type	Reset
31:0	hash_cpa_en	Enable CPA for each hashed HNF node ID	RW	64'h0000000000000000

hashed_target_grp_hnf_cpa_en_reg_2-1

This register repeats -1 times. It parametrized by the index from 2 to 1. Configures CCIX port aggregation mode for hashed HNF node IDs

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1180
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

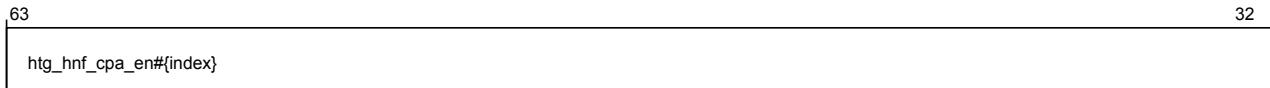


Figure 3-603 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (high)

The following table shows the hashed_target_grp_hnf_cpa_en_reg_2-1 higher register bit assignments.

Table 3-623 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (high)

Bits	Field name	Description	Type	Reset
63:32	htg_hnf_cpa_en#{index}	<p>Enable CPA for each hashed HNF node ID</p>	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

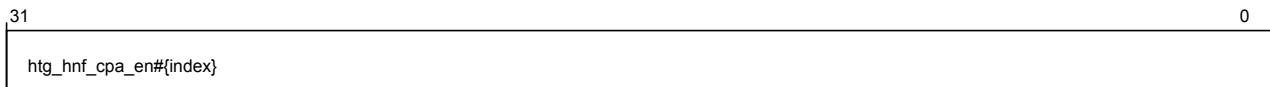


Figure 3-604 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (low)

The following table shows the hashed_target_grp_hnf_cpa_en_reg_2-1 lower register bit assignments.

Table 3-624 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (low)

Bits	Field name	Description	Type	Reset
31:0	htg_hnf_cpa_en#{index}	<p>Enable CPA for each hashed HNF node ID</p>	RW	64'h0000000000000000

hashed_target_grp_hnf_cpa_en_reg_2-1

This register repeats -1 times. It parametrized by the index from 2 to 1. Configures CCIX port aggregation mode for hashed HNF node IDs

Its characteristics are:

Type	RW
Register width (Bits)	64

Address offset	16'h1180
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

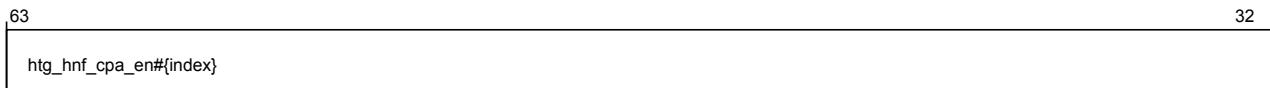


Figure 3-605 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (high)

The following table shows the hashed_target_grp_hnf_cpa_en_reg_2-1 higher register bit assignments.

Table 3-625 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (high)

Bits	Field name	Description	Type	Reset
63:32	htg_hnf_cpa_en#{index}	<p>Enable CPA for each hashed HNF node ID</p>	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

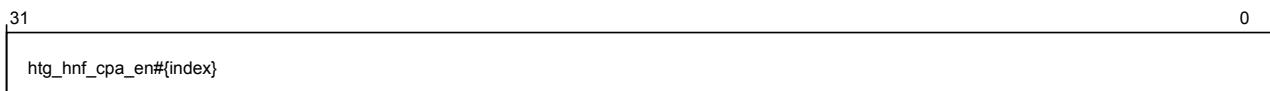


Figure 3-606 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (low)

The following table shows the hashed_target_grp_hnf_cpa_en_reg_2-1 lower register bit assignments.

Table 3-626 por_rnsam_hashed_target_grp_hnf_cpa_en_reg_2-1 (low)

Bits	Field name	Description	Type	Reset
31:0	htg_hnf_cpa_en#{index}	<p>Enable CPA for each hashed HNF node ID</p>	RW	64'h0000000000000000

hashed_target_grp_hnp_cpa_en_reg_0-0

This register repeats 0 times. It parametrized by the index from 0 to 0. Configures CCIX port aggregation mode for hashed HNF node IDs

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3730 + #{8 * [0]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63

32

hash_hnp_cpa_en#{index}

Figure 3-607 por_rnsam_hashed_target_grp_hnp_cpa_en_reg_0-0 (high)

The following table shows the hashed_target_grp_hnp_cpa_en_reg_0-0 higher register bit assignments.

Table 3-627 por_rnsam_hashed_target_grp_hnp_cpa_en_reg_0-0 (high)

Bits	Field name	Description	Type	Reset
63:32	hash_hnp_cpa_en#{index}	Enable CPA for each hashed HNP node ID	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

31

0

hash_hnp_cpa_en#{index}

Figure 3-608 por_rnsam_hashed_target_grp_hnp_cpa_en_reg_0-0 (low)

The following table shows the hashed_target_grp_hnp_cpa_en_reg_0-0 lower register bit assignments.

Table 3-628 por_rnsam_hashed_target_grp_hnp_cpa_en_reg_0-0 (low)

Bits	Field name	Description	Type	Reset
31:0	hash_hnp_cpa_en#{index}	Enable CPA for each hashed HNP node ID	RW	64'h0000000000000000

sys_cache_grp_hn_cpa_grp_reg

Configures CCIX port aggregation group ID for each System Cache Group

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1190

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

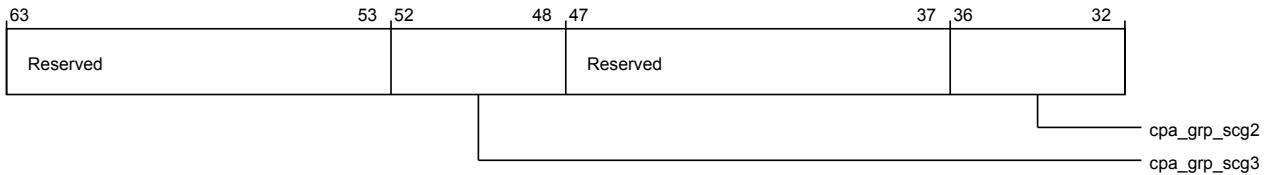


Figure 3-609 por_rnsam_sys_cache_grp_hn_cpa_grp_reg (high)

The following table shows the sys_cache_grp_hn_cpa_grp_reg higher register bit assignments.

Table 3-629 por_rnsam_sys_cache_grp_hn_cpa_grp_reg (high)

Bits	Field name	Description	Type	Reset
63:53	Reserved	Reserved	RO	-
52:48	cpa_grp_scg3	Specifies CCIX port aggregation group ID for Hashed Target Group 3	RW	5'h0
47:37	Reserved	Reserved	RO	-
36:32	cpa_grp_scg2	Specifies CCIX port aggregation group ID for Hashed target Group 2	RW	5'h0

The following image shows the lower register bit assignments.

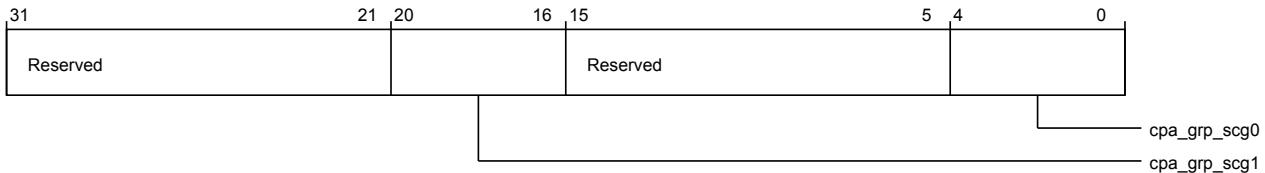


Figure 3-610 por_rnsam_sys_cache_grp_hn_cpa_grp_reg (low)

The following table shows the sys_cache_grp_hn_cpa_grp_reg lower register bit assignments.

Table 3-630 por_rnsam_sys_cache_grp_hn_cpa_grp_reg (low)

Bits	Field name	Description	Type	Reset
31:21	Reserved	Reserved	RO	-
20:16	cpa_grp_scg1	Specifies CCIX port aggregation group ID for Hashed target Group 1	RW	5'h0
15:5	Reserved	Reserved	RO	-
4:0	cpa_grp_scg0	Specifies CCIX port aggregation group ID for hashed target Group 0	RW	5'h0

hashed_target_grp_cpa_grp_reg_2-7

This register repeats 5 times. It parametrized by the index from 2 to 7. Configures CCIX port aggregation group ID for each System Cache Group

Its characteristics are:

Type	RW
------	----

Register width (Bits) 64

Address offset 16'h1190

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

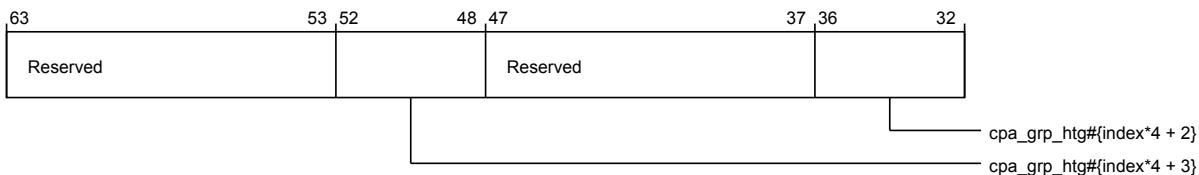


Figure 3-611 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (high)

The following table shows the hashed_target_grp_cpa_grp_reg_2-7 higher register bit assignments.

Table 3-631 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (high)

Bits	Field name	Description	Type	Reset
63:53	Reserved	Reserved	RO	-
52:48	cpa_grp_htg#{index*4 + 3}	<p>Specifies CCIX port aggregation group ID for Hashed Target Group #{index*4 + 3}</p>	RW	5'h0
47:37	Reserved	Reserved	RO	-
36:32	cpa_grp_htg#{index*4 + 2}	<p>Specifies CCIX port aggregation group ID for Hashed target Group #{index*4 + 2}</p>	RW	5'h0

The following image shows the lower register bit assignments.

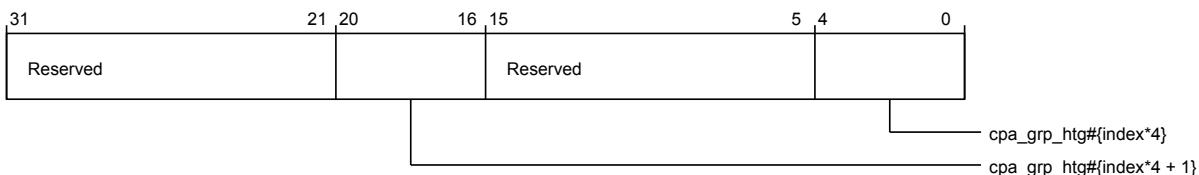


Figure 3-612 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (low)

The following table shows the hashed_target_grp_cpa_grp_reg_2-7 lower register bit assignments.

Table 3-632 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (low)

Bits	Field name	Description	Type	Reset
31:21	Reserved	Reserved	RO	-
20:16	cpa_grp_htg#{index*4 + 1}	<p>Specifies CCIX port aggregation group ID for Hashed target Group #{index*4 + 1}</p>	RW	5'h0
15:5	Reserved	Reserved	RO	-
4:0	cpa_grp_htg#{index*4}	<p>Specifies CCIX port aggregation group ID for hashed target Group #{index*4}</p>	RW	5'h0

hashed_target_grp_cpa_grp_reg_2-7

This register repeats 5 times. It parametrized by the index from 2 to 7. Configures CCIX port aggregation group ID for each System Cache Group

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1190

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

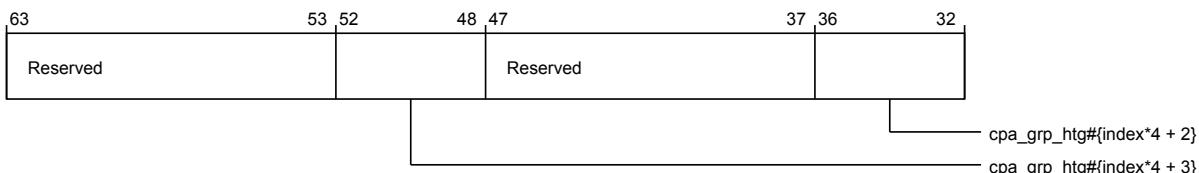


Figure 3-613 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (high)

The following table shows the hashed_target_grp_cpa_grp_reg_2-7 higher register bit assignments.

Table 3-633 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (high)

Bits	Field name	Description	Type	Reset
63:53	Reserved	Reserved	RO	-
52:48	cpa_grp_htg#{index*4 + 3}	<p>Specifies CCIX port aggregation group ID for Hashed Target Group #{index*4 + 3}</p>	RW	5'h0
47:37	Reserved	Reserved	RO	-
36:32	cpa_grp_htg#{index*4 + 2}	<p>Specifies CCIX port aggregation group ID for Hashed target Group #{index*4 + 2}</p>	RW	5'h0

The following image shows the lower register bit assignments.

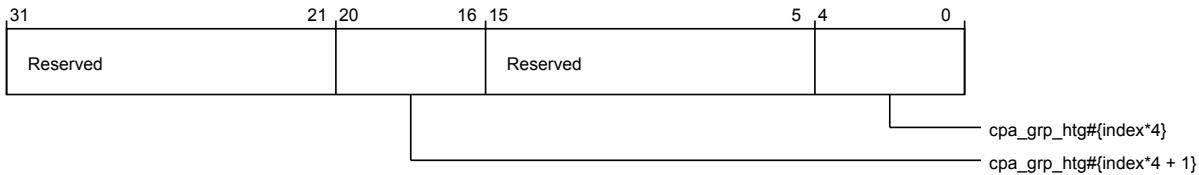


Figure 3-614 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (low)

The following table shows the hashed_target_grp_cpa_grp_reg_2-7 lower register bit assignments.

Table 3-634 por_rnsam_hashed_target_grp_cpa_grp_reg_2-7 (low)

Bits	Field name	Description	Type	Reset
31:21	Reserved	Reserved	RO	-
20:16	cpa_grp_htg#{index*4 + 1}	<p>Specifies CCIX port aggregation group ID for Hashed target Group #{index*4 + 1}</p>	RW	5'h0
15:5	Reserved	Reserved	RO	-
4:0	cpa_grp_htg#{index*4}	<p>Specifies CCIX port aggregation group ID for hashed target Group #{index*4}</p>	RW	5'h0

hashed_target_grp_hnf_device_bound_cfg_reg_0-1

This register repeats 1 times. It parametrized by the index from 0 to 1. Configures cache lines routed to this HNF as Device bound or Home bound 1'b0: cache lines routed to Home bound 1'b1: cache lines routed to Device bound

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h37C0 + #{8 * [0, 1]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

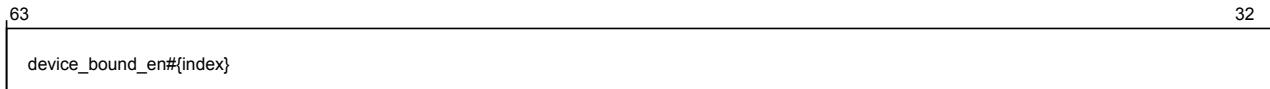


Figure 3-615 por_rnsam_hashed_target_grp_hnf_device_bound_cfg_reg_0-1 (high)

The following table shows the hashed_target_grp_hnf_device_bound_cfg_reg_0-1 higher register bit assignments.

Table 3-635 por_rnsam_hashed_target_grp_hnf_device_bound_cfg_reg_0-1 (high)

Bits	Field name	Description	Type	Reset
63:32	device_bound_en#{index}	Enable Device bound configuration for hashed HNF node ID	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

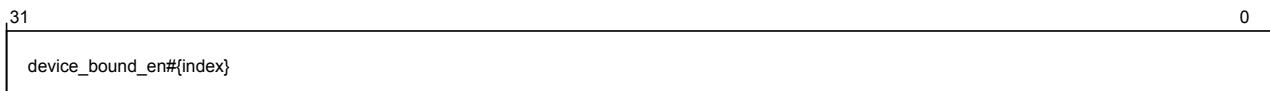


Figure 3-616 por_rnsam_hashed_target_grp_hnf_device_bound_cfg_reg_0-1 (low)

The following table shows the hashed_target_grp_hnf_device_bound_cfg_reg_0-1 lower register bit assignments.

Table 3-636 por_rnsam_hashed_target_grp_hnf_device_bound_cfg_reg_0-1 (low)

Bits	Field name	Description	Type	Reset
31:0	device_bound_en#{index}	Enable Device bound configuration for hashed HNF node ID	RW	64'h0000000000000000

rnsam_hash_addr mask reg

Configures the address mask that is applied before hashing the address bits.

Its characteristics are:

Type

Register width (Bits) 64

Address offset 16 hE80

Register reset	64'b11
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-617 por rnsam rnsam hash addr mask reg (high)

The following table shows the rnsam hash addr mask reg higher register bit assignments.

Table 3-637 por_rnsam_rnsam_hash_addr_mask_reg (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	Address mask applied before hashing	RW	46'h3FFFFFFFFFFF

The following image shows the lower register bit assignments.

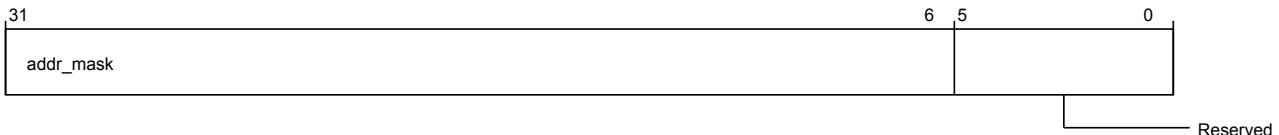


Figure 3-618 por_rnsam_rnsam_hash_addr_mask_reg (low)

The following table shows the rnsam_hash_addr_mask_reg lower register bit assignments.

Table 3-638 por_rnsam_rnsam_hash_addr_mask_reg (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask	Address mask applied before hashing	RW	46'h3FFFFFFFFFFF
5:0	Reserved	Reserved	RO	-

rnsam_hash_axi_id_mask_reg

Configures the AXI ID mask that is applied before hashing the address bits.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hE88

The following image shows the higher register bit assignments

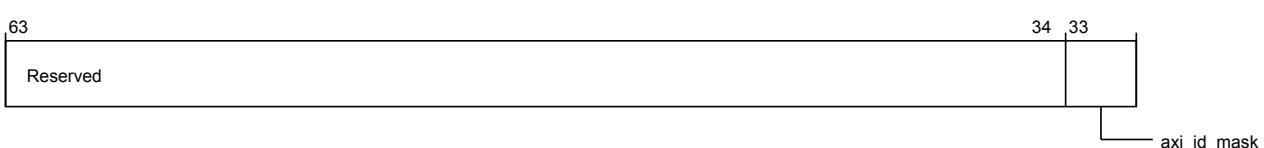


Figure 3-619 por_rnsam_rnsam_hash_axi_id_mask_reg (high)

The following table shows the rnsam hash axi id mask reg higher register bit assignments.

Table 3-639 por_rnsam_rnsam_hash_axi_id_mask_reg (high)

Bits	Field name	Description	Type	Reset
63:34	Reserved	Reserved	RO	-
33:32	axi_id_mask	AXI_ID mask applied before hashing	RW	34'h3FFFFFFF

The following image shows the lower register bit assignments.



Figure 3-620 por_rnsam_rnsam_hash_axi_id_mask_reg (low)

The following table shows the rnsam_hash_axi_id_mask_reg lower register bit assignments.

Table 3-640 por_rnsam_rnsam_hash_axi_id_mask_reg (low)

Bits	Field name	Description	Type	Reset
31:0	axi_id_mask	AXI_ID mask applied before hashing	RW	34'hxFFFFFFFF

rnsam_region_cmp_addr_mask_reg

Configures the address mask that is applied before region compare.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hE90

Register reset	64'b11111111111111111111111111111111
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments



Figure 3-621 por_rnsam_rnsam_region_cmp_addr_mask_reg (high)

The following table shows the rnsam region cmp addr mask reg higher register bit assignments.

Table 3-641 por_rnsam_rnsam_region_cmp_addr_mask_reg (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	Address mask applied before memory region compare	RW	36'hFFFFFFFF

The following image shows the lower register bit assignments.

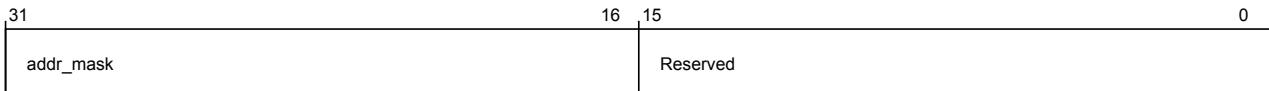


Figure 3-622 por_rnsam_rnsam_region_cmp_addr_mask_reg (low)

The following table shows the rnsam_region, cmp, addr, mask, reg, lower register bit assignments.

Table 3-642 por_rnsam_rnsam_region_cmp_addr_mask_reg (low)

Bits	Field name	Description	Type	Reset
31:16	addr_mask	Address mask applied before memory region compare	RW	36'hFFFFFFFF
15:0	Reserved	Reserved	RO	-

cml_port_aggr_grp_6-31_add_mask

This register repeats 25 times. It parametrized by the index from 6 to 31. Configures the CCIX port aggregation address mask for group #*{index}*.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

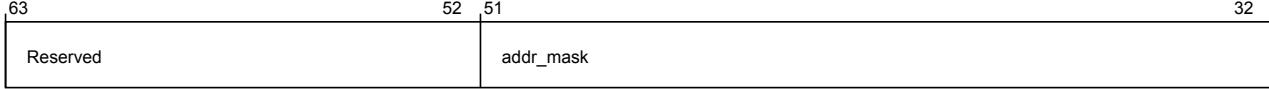


Figure 3-623 por_rnsam_cml_port_aggr_grp_6-31_add_mask (high)

The following table shows the cml_port_aggr_grp_6-31_add_mask higher register bit assignments.

Table 3-643 por_rnsam_cml_port_aggr_grp_6-31_add_mask (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	<p>Address mask to be applied before hashing</p> <p>CONSTRAINT: ADDR MASK is [51:6]</p>	RW	46'h3FFFFFFFFFFF

The following image shows the lower register bit assignments.

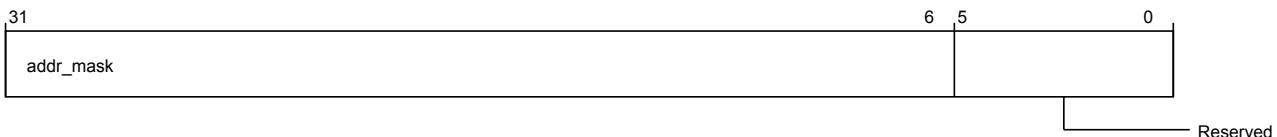


Figure 3-624 por_rnsam_cml_port_aggr_grp_6-31_add_mask (low)

The following table shows the cml_port_aggr_grp_6-31 add mask lower register bit assignments.

Table 3-644 por_rnsam_cml_port_aggr_grp_6-31_add_mask (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask	<p>Address mask to be applied before hashing</p> <p>CONSTRAINT: ADDR MASK is [51:6]</p>	RW	46'hxFFFFFFFFFFFF
5:0	Reserved	Reserved	RO	-

cml_port_aggr_grp_6-31_add_mask

This register repeats 25 times. It parametrized by the index from 6 to 31. Configures the CCIX port aggregation address mask for group #*{index}*.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset

The following image shows the higher register bit assignments

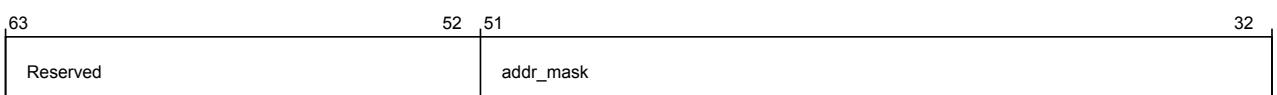


Figure 3-625 por rnsam cml port aqqr grp 6-31 add mask (high)

The following table shows the cml_port_aggr_grp_6-31_add_mask_higher_register_bit assignments.

Table 3-645 por_rnsam_cml_port_aggr_grp_6-31_add_mask (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	<p>Address mask to be applied before hashing</p> <p>CONSTRAINT: ADDR MASK is [51:6]</p>	RW	46'h3FFFFFFFFF

The following image shows the lower register bit assignments.

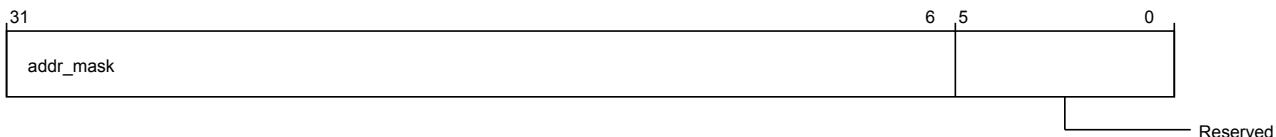


Figure 3-626 por_rnsam_cml_port_aggr_grp_6-31_add_mask (low)

The following table shows the cml_port_aggr_grp_6-31_add_mask lower register bit assignments.

Table 3-646 por_rnsam_cml_port_aggr_grp_6-31_add_mask (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask	<p>Address mask to be applied before hashing</p> <p>CONSTRAINT: ADDR MASK is [51:6]</p>	RW	46'h3FFFFFFFFF
5:0	Reserved	Reserved	RO	-

cml_port_aggr_grp_reg_3-12

This register repeats 9 times. It parametrized by the index from 3 to 12. Configures the CCIX port aggregation port Node IDs

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h11F0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

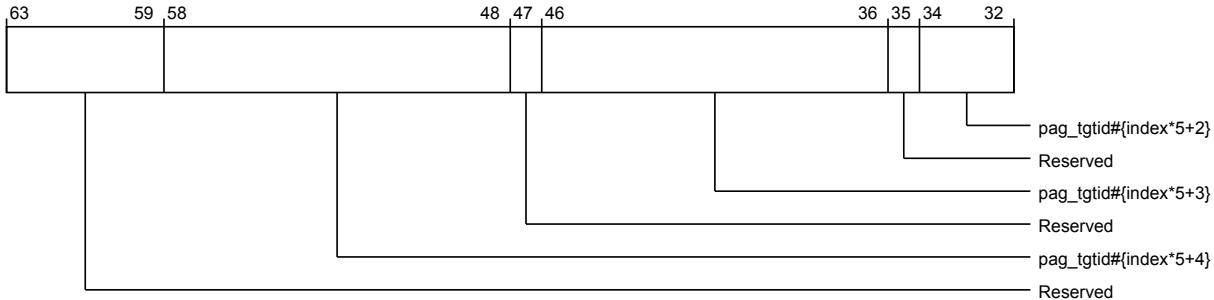


Figure 3-627 por_rnsam_cml_port_aggr_grp_reg_3-12 (high)

The following table shows the cml_port_aggr_grp_reg_3-12 higher register bit assignments.

Table 3-647 por_rnsam_cml_port_aggr_grp_reg_3-12 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	pag_tgtid#{index*5+4}	<p>Specifies target ID #{index*5+4} for CPAG</p>	RW	11'b0
47	Reserved	Reserved	RO	-
46:36	pag_tgtid#{index*5+3}	<p>Specifies target ID #{index*5+3} for CPAG</p>	RW	11'b0
35	Reserved	Reserved	RO	-
34:32	pag_tgtid#{index*5+2}	<p>Specifies target ID #{index*5+2} for CPAG</p>	RW	11'b0

The following image shows the lower register bit assignments.

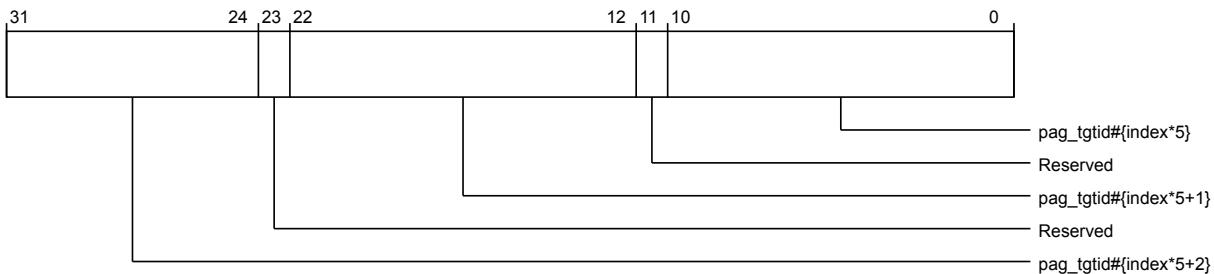


Figure 3-628 por_rnsam_cml_port_aggr_grp_reg_3-12 (low)

The following table shows the cml port aggr grp reg 3-12 lower register bit assignments.

Table 3-648 por_rnsam_cml_port_aggr_grp_reg_3-12 (low)

Bits	Field name	Description	Type	Reset
31:24	pag_tgtid#{index*5+2}	<p>Specifies target ID #{index*5+2} for CPAG</p>	RW	11'b0
23	Reserved	Reserved	RO	-
22:12	pag_tgtid#{index*5+1}	<p>Specifies target ID #{index*5+1} for CPAG</p>	RW	11'b0

Table 3-648 por_rnsam_cml_port_aggr_grp_reg_3-12 (low) (continued)

Bits	Field name	Description	Type	Reset
11	Reserved	Reserved	RO	-
10:0	pag_tgtid#{index*5}	<p>Specifies target ID #{index*5} for CPAG</p>	RW	11'b0

cml_port_aggr_grp_reg_3-12

This register repeats 9 times. It parametrized by the index from 3 to 12. Configures the CCIX port aggregation port Node IDs

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1

Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments

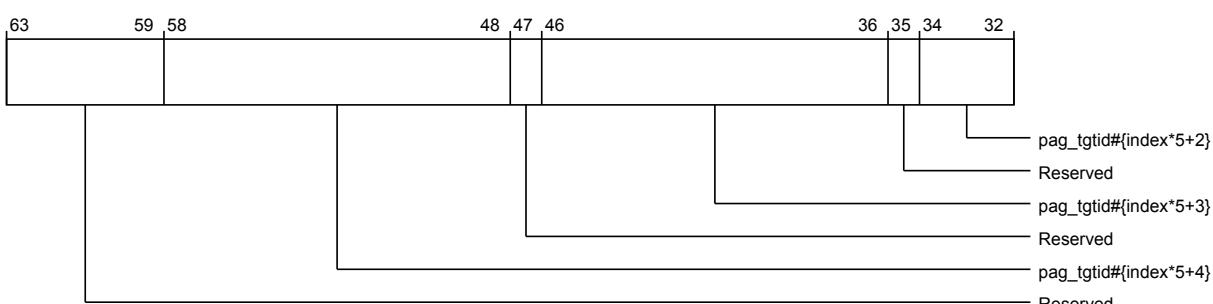


Figure 3-629 por rnsam cml port aggr grp req 3-12 (high)

The following table shows the cml_port_aggr_grp_reg_3-12 higher register bit assignments.

Table 3-649 por rnsam cml port aggr grp req 3-12 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	pag_tgtid#{index*5+4}	<p>Specifies target ID #{index*5+4} for CPAG</p>	RW	11'b0
47	Reserved	Reserved	RO	-
46:36	pag_tgtid#{index*5+3}	<p>Specifies target ID #{index*5+3} for CPAG</p>	RW	11'b0
35	Reserved	Reserved	RO	-
34:32	pag_tgtid#{index*5+2}	<p>Specifies target ID #{index*5+2} for CPAG</p>	RW	11'b0

The following image shows the lower register bit assignments.

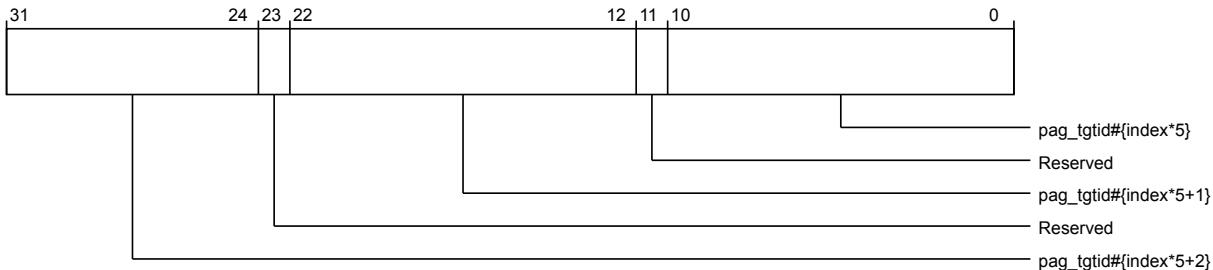


Figure 3-630 por_rnsam_cml_port_aggr_grp_reg_3-12 (low)

The following table shows the cml_port_aggr_grp_reg_3-12 lower register bit assignments.

Table 3-650 por_rnsam_cml_port_aggr_grp_reg_3-12 (low)

Bits	Field name	Description	Type	Reset
31:24	pag_tgtid#{index*5+2}	<p>Specifies target ID #{index*5+2} for CPAG</p>	RW	11'b0
23	Reserved	Reserved	RO	-
22:12	pag_tgtid#{index*5+1}	<p>Specifies target ID #{index*5+1} for CPAG</p>	RW	11'b0
11	Reserved	Reserved	RO	-
10:0	pag_tgtid#{index*5}	<p>Specifies target ID #{index*5} for CPAG</p>	RW	11'b0

cml_port_aggr_ctrl_reg

Configures the CCIX port aggregation port IDs for group 2.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1208
Register reset	64'b1010101010
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

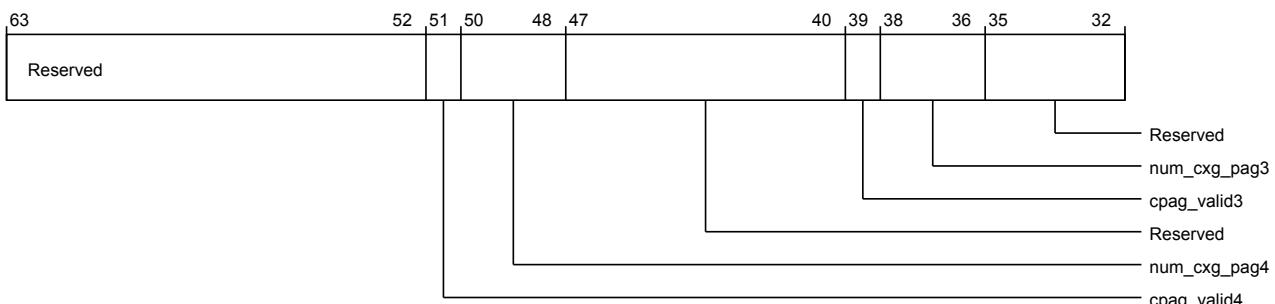


Figure 3-631 por_rnsam_cml_port_aggr_ctrl_reg (high)

The following table shows the cml_port_aggr_ctrl_reg higher register bit assignments.

Table 3-651 por_rnsam_cml_port_aggr_ctrl_reg (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51	cpag_valid4	Valid programming for CPAG4, Enabled by default (backward compatible)	RW	1'b1
50:48	num_cxg_pag4	Specifies the number of CXRAs in CPAG4 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0
47:40	Reserved	Reserved	RO	-
39	cpag_valid3	Valid programming for CPAG + 3}, Enabled by default (backward compatible)	RW	1'b1
38:36	num_cxg_pag3	Specifies the number of CXRAs in CPAG3 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0
35:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

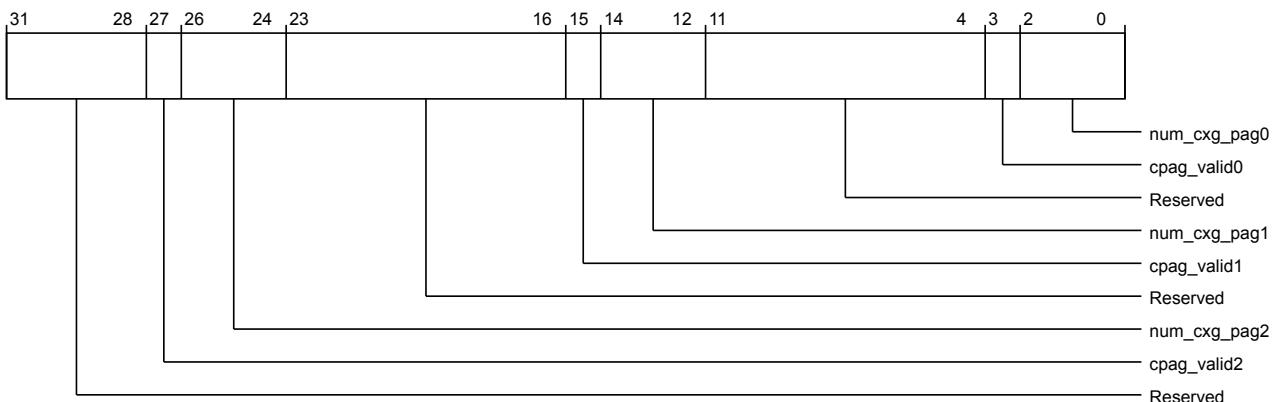


Figure 3-632 por_rnsam_cml_port_aggr_ctrl_reg (low)

The following table shows the cml port aggr ctrl reg lower register bit assignments.

Table 3-652 por_rnsam_cml_port_aggr_ctrl_reg (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27	cpag_valid2	Valid programming for CPAG + 2}, Enabled by default (backward compatible)	RW	1'b1
26:24	num_cxg_pag2	Specifies the number of CXRAs in CPAG2 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0
23:16	Reserved	Reserved	RO	-
15	cpag_valid1	Valid programming for CPAG + 1}, Enabled by default (backward compatible)	RW	1'b1
14:12	num_cxg_pag1	Specifies the number of CXRAs in CPAG1 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0
11:4	Reserved	Reserved	RO	-
3	cpag_valid0	Valid programming for CPAG, Enabled by default (backward compatible)	RW	1'b1
2:0	num_cxg_pag0	Specifies the number of CXRAs in CPAG0 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0

cml_port_aggr_ctrl_reg_0-6

This register repeats 6 times. It parametrized by the index from 0 to 6. Configures the CCIX port aggregation port IDs for group 2.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1208 + #{8 * [0, 1, 2, 3, 4, 5, 6]}
Register reset	64'b1010101010
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

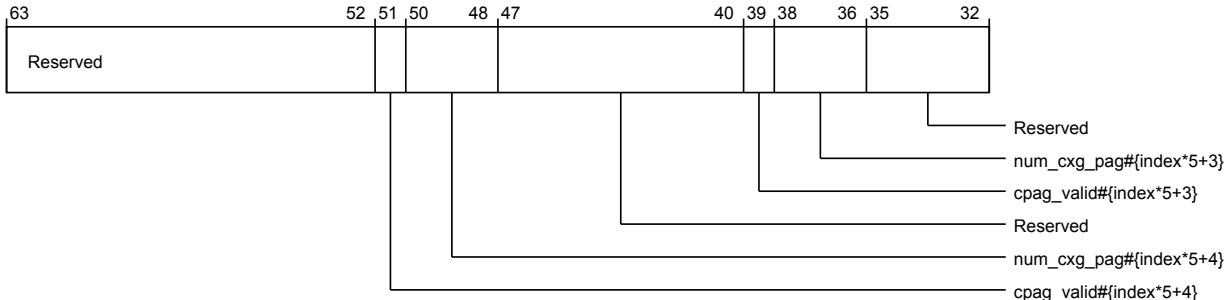


Figure 3-633 por_rnsam_cml_port_aggr_ctrl_reg_0-6 (high)

The following table shows the cml port aggr ctrl reg 0-6 higher register bit assignments.

Table 3-653 por_rnsam_cml_port_aggr_ctrl_reg_0-6 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51	cpag_valid#{index*5+4}	Valid programming for CPAG #{index*5 + 4}, Enabled by default (backward compatible)	RW	1'b1
50:48	num_cxg_pag#{index*5+4}	Specifies the number of CXRAs in CPAG #{index*5 + 4} 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0
47:40	Reserved	Reserved	RO	-
39	cpag_valid#{index*5+3}	Valid programming for CPAG #{index*5 + 3}, Enabled by default (backward compatible)	RW	1'b1

Table 3-653 por_rnsam_cml_port_aggr_ctrl_reg_0-6 (high) (continued)

Bits	Field name	Description	Type	Reset
38:36	num_cxg_pag#{index*5+3}	Specifies the number of CXRAs in CPAG #{@{index*5 + 3}} 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b00
35:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

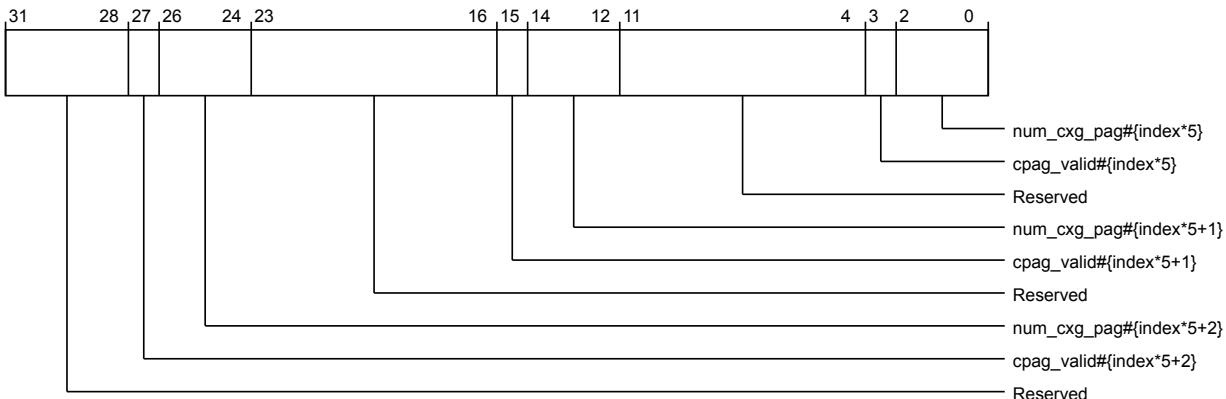


Figure 3-634 por_rnsam_cml_port_aggr_ctrl_reg_0-6 (low)

The following table shows the cml port aggr ctrl reg 0-6 lower register bit assignments.

Table 3-654 por_rnsam_cml_port_aggr_ctrl_reg_0-6 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27	cpag_valid#{index*5+2}	Valid programming for CPAG # $\{index * 5 + 2\}$, Enabled by default (backward compatible)	RW	1'b1

Table 3-654 por_rnsam_cml_port_aggr_ctrl_reg_0-6 (low) (continued)

Bits	Field name	Description	Type	Reset
26:24	num_cxg_pag#{index*5+2}	Specifies the number of CXRAs in CPAG #{index*5 + 2} 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0
23:16	Reserved	Reserved	RO	-
15	cpag_valid#{index*5+1}	Valid programming for CPAG #{index*5 + 1}, Enabled by default (backward compatible)	RW	1'b1
14:12	num_cxg_pag#{index*5+1}	Specifies the number of CXRAs in CPAG #{index*5 + 1} 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0
11:4	Reserved	Reserved	RO	-
3	cpag_valid#{index*5}	Valid programming for CPAG #{index*5}, Enabled by default (backward compatible)	RW	1'b1
2:0	num_cxg_pag#{index*5}	Specifies the number of CXRAs in CPAG #{index*5} 3'b000: 1 port 3'b001: 2 ports 3'b010: 4 ports 3'b011: 8 ports 3'b100: 16 ports 3'b101: 32 ports others: Reserved	RW	3'b0

sys_cache_grp_sn_attr

Configures attributes for SN node IDs for system cache groups.

Its characteristics are:

Type	RW
-------------	----

Register width (Bits)	64
Address offset	16'hEB0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

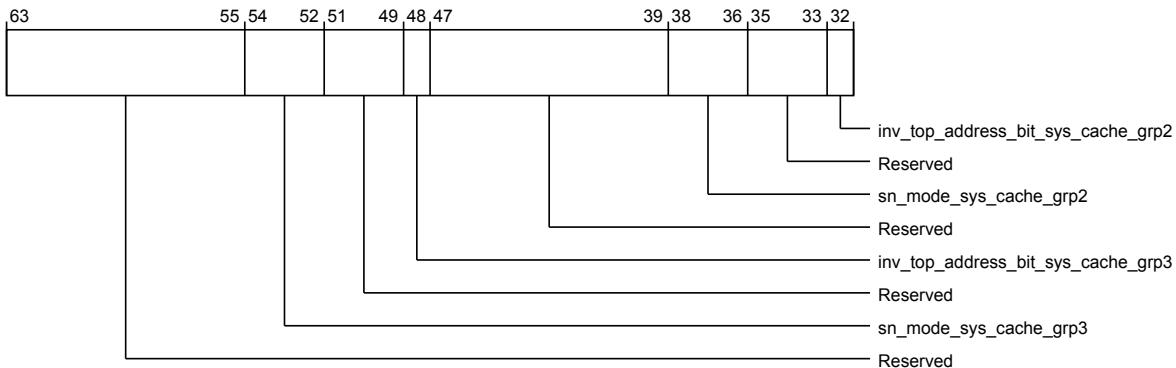


Figure 3-635 por_rnsam_sys_cache_grp_sn_attr (high)

The following table shows the sys_cache_grp_sn_attr higher register bit assignments.

Table 3-655 por_rnsam_sys_cache_grp_sn_attr (high)

Bits	Field name	Description	Type	Reset
63:55	Reserved	Reserved	RO	-
54:52	sn_mode_sys_cache_grp3	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b0
51:49	Reserved	Reserved	RO	-
48	inv_top_address_bit_sys_cache_grp3	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
47:39	Reserved	Reserved	RO	-
38:36	sn_mode_sys_cache_grp2	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b00

Table 3-655 por_rnsam_sys_cache_grp_sn_attr (high) (continued)

Bits	Field name	Description	Type	Reset
35:33	Reserved	Reserved	RO	-
32	inv_top_address_bit_sys_cache_grp2	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0

The following image shows the lower register bit assignments.

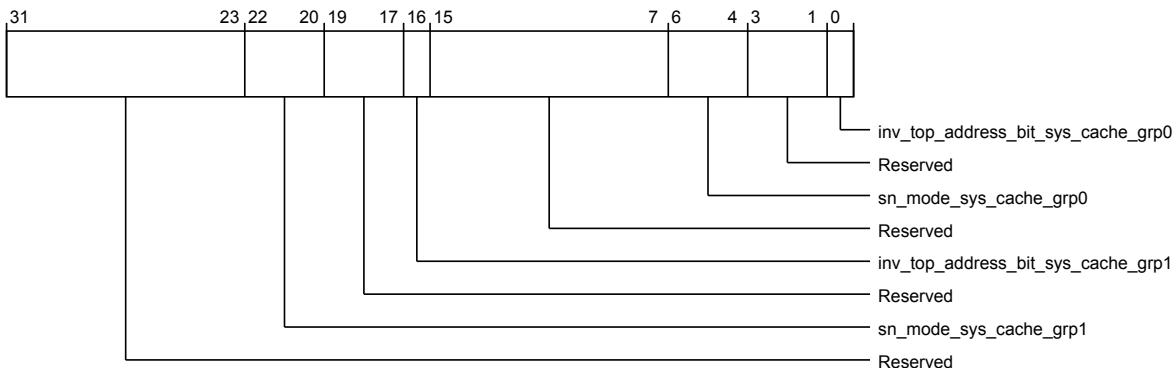


Figure 3-636 por_rnsam_sys_cache_grp_sn_attr (low)

The following table shows the sys_cache_grp_sn_attr lower register bit assignments.

Table 3-656 por_rnsam_sys_cache_grp_sn_attr (low)

Bits	Field name	Description	Type	Reset
31:23	Reserved	Reserved	RO	-
22:20	sn_mode_sys_cache_grp1	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b0
19:17	Reserved	Reserved	RO	-
16	inv_top_address_bit_sys_cache_grp1	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
15:7	Reserved	Reserved	RO	-

Table 3-656 por_rnsam_sys_cache_grp_sn_attr (low) (continued)

Bits	Field name	Description	Type	Reset
6:4	sn_mode_sys_cache_grp0	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b0
3:1	Reserved	Reserved	RO	-
0	inv_top_address_bit_sys_cache_grp0	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0

sys_cache_grp_sn_attr1

Configures attributes for SN node IDs for system cache groups.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hEB8

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-637 por_rnsam_sys_cache_grp_sn_attr1 (high)

The following table shows the sys_cache_grp_sn_attr1 higher register bit assignments.

Table 3-657 por_rnsam_sys_cache_grp_sn_attr1 (high)

Bits	Field name	Description	Type	Reset
63:55	Reserved	Reserved	RO	-
54:52	sn_mode_sys_cache_grp7	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b0
51:49	Reserved	Reserved	RO	-
48	inv_top_address_bit_sys_cache_grp7	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
47:39	Reserved	Reserved	RO	-
38:36	sn_mode_sys_cache_grp6	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b00
35:33	Reserved	Reserved	RO	-
32	inv_top_address_bit_sys_cache_grp6	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0

The following image shows the lower register bit assignments.

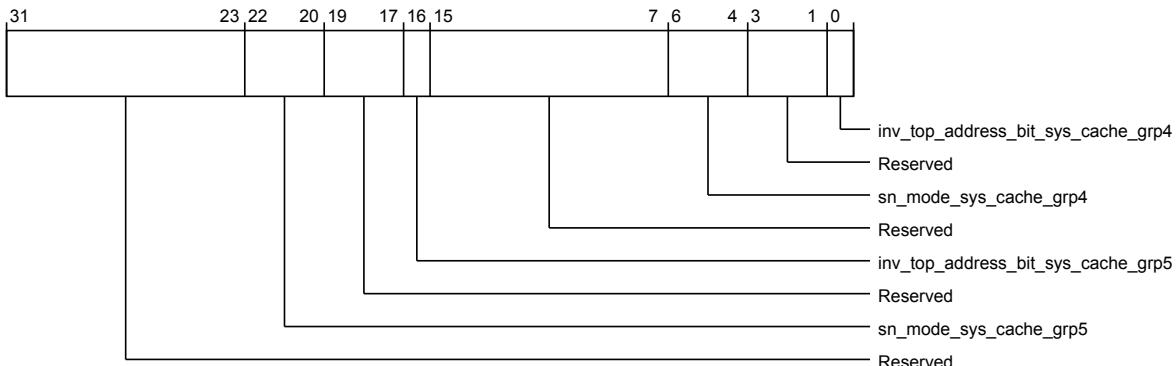


Figure 3-638 por_rnsam_sys_cache_grp_sn_attr1 (low)

The following table shows the sys_cache_grp_sn_attr1 lower register bit assignments.

Table 3-658 por_rnsam_sys_cache_grp_sn_attr1 (low)

Bits	Field name	Description	Type	Reset
31:23	Reserved	Reserved	RO	-
22:20	sn_mode_sys_cache_grp5	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b0
19:17	Reserved	Reserved	RO	-
16	inv_top_address_bit_sys_cache_grp5	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
15:7	Reserved	Reserved	RO	-
6:4	sn_mode_sys_cache_grp4	SN selection mode 3'b000: 1-SN mode (SN0) 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) Others: Reserved	RW	3'b0
3:1	Reserved	Reserved	RO	-
0	inv_top_address_bit_sys_cache_grp4	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0

sys_cache_grp_sn_sam_cfg_0-3

This register repeats 3 times. It parametrized by the index from 0 to 3. Configures top address bits for SN SAM system cache groups #{{index}*2} and #{{index}*2 + 1}. All top_address_bit fields must be between bits 47 and 28. top_address_bit2 > top_address_bit1 > top_address_bit0.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1140 + #{{8 * [0, 1, 2, 3]}}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

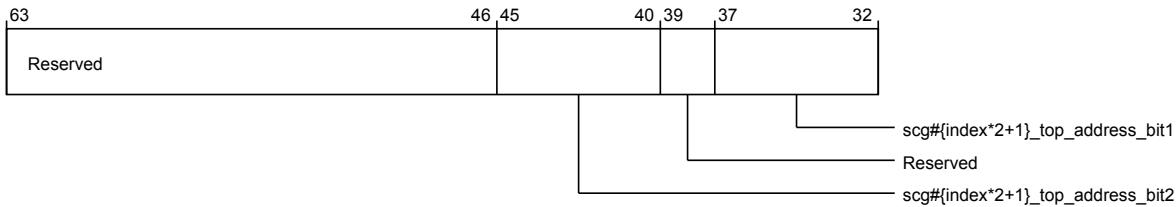


Figure 3-639 por_rnsam_sys_cache_grp_sn_sam_cfg_0-3 (high)

The following table shows the sys_cache_grp_sn_sam_cfg_0-3 higher register bit assignments.

Table 3-659 por_rnsam_sys_cache_grp_sn_sam_cfg_0-3 (high)

Bits	Field name	Description	Type	Reset
63:46	Reserved	Reserved	RO	-
45:40	scg#{index*2+1}_top_address_bit2	Top address bit 2 for system cache group #{index*2+1}	RW	6'h00
39:38	Reserved	Reserved	RO	-
37:32	scg#{index*2+1}_top_address_bit1	Top address bit 1 for system cache group #{index*2+1}	RW	6'h00

The following image shows the lower register bit assignments.

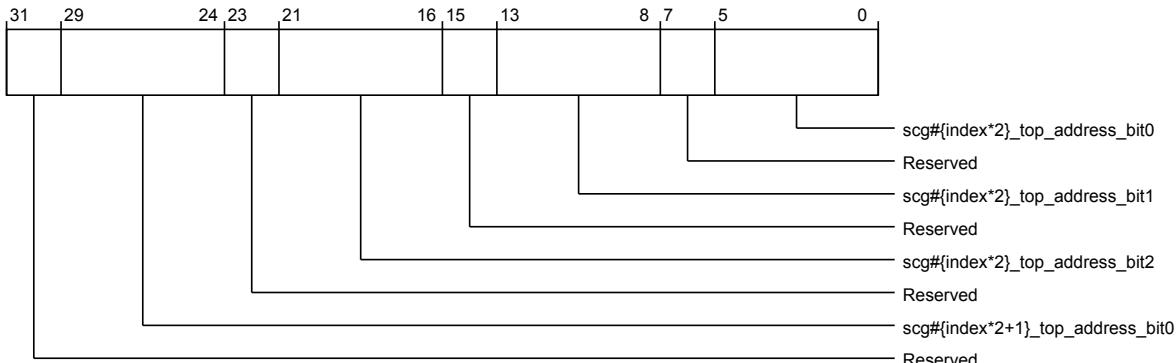


Figure 3-640 por_rnsam_sys_cache_grp_sn_sam_cfg_0-3 (low)

The following table shows the sys_cache_grp_sn_sam_cfg_0-3 lower register bit assignments.

Table 3-660 por_rnsam_sys_cache_grp_sn_sam_cfg_0-3 (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	scg#{index*2+1}_top_address_bit0	Top address bit 0 for system cache group #{index*2+1}	RW	6'h00
23:22	Reserved	Reserved	RO	-
21:16	scg#{index*2}_top_address_bit2	Top address bit 2 for system cache group #{index*2}	RW	6'h00
15:14	Reserved	Reserved	RO	-

Table 3-660 por_rnsam_sys_cache_grp_sn_sam_cfg_0-3 (low) (continued)

Bits	Field name	Description	Type	Reset
13:8	scg#{index*2}_top_address_bit1	Top address bit 1 for system cache group #{index*2}	RW	6'h00
7:6	Reserved	Reserved	RO	-
5:0	scg#{index*2}_top_address_bit0	Top address bit 0 for system cache group #{index*2}	RW	6'h00

sam_qos_mem_region_reg_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Configures the QoS value for memory region #*{index}*

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h17

Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first access to the corresponding memory location.

Secure group por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

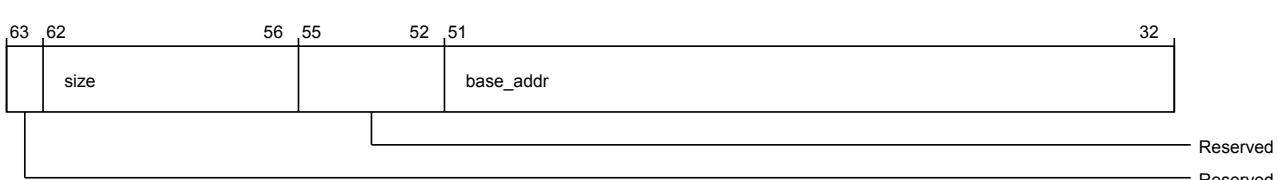


Figure 3-641 por rnsam sam qos mem region req 0-15 (high)

The following table shows the same QoS, mem, region, reg_0-15 higher register bit assignments.

Table 3-661 por rnsam sam gos mem region req 0-15 (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:56	size	Memory region size CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).	RW	5'b00000
55:52	Reserved	Reserved	RO	-
51:32	base_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0

The following image shows the lower register bit assignments.

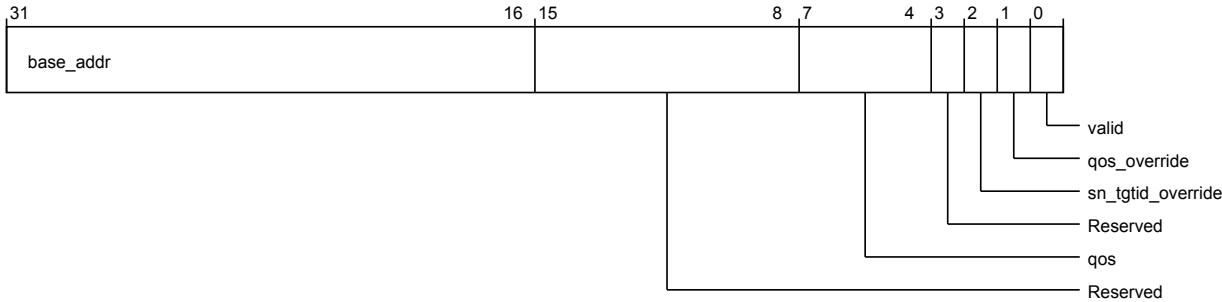


Figure 3-642 por_rnsam_sam_qos_mem_region_reg_0-15 (low)

The following table shows the sam_qos_mem_region_reg_0-15 lower register bit assignments.

Table 3-662 por_rnsam_sam_qos_mem_region_reg_0-15 (low)

Bits	Field name	Description	Type	Reset
31:16	base_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0
15:8	Reserved	Reserved	RO	-
7:4	qos	Indicates the QoS value to be used for this region	RW	4'b0000
3	Reserved	Reserved	RO	-
2	sn_tgtid_override	Override the SN targetId for address contained in the region of this register	RW	1'b0
1	qos_override	QoS Memory region allow override 1'b0: Do not override the QoS value from the QoS regulator 1'b1: Override the QoS value with the programmed value in regionX_qos	RW	1'b0
0	valid	QoS Memory region valid 1'b0: Not valid 1'b1: Valid for memory region comparison	RW	1'b0

sam_qos_mem_region_cfg2_reg_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Configures the QOS memory region #*{index}*

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h11

Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the

Secure group por_rnsam_secure_register_groups_override.mem_range
first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63	52	51	32
Reserved		end_addr	

Figure 3-643 por_rnsam_sam_qos_mem_region_cfg2_reg_0-15 (high)

The following table shows the sam_qos_mem_region_cfg2_reg_0-15 higher register bit assignments.

Table 3-663 por_rnsam_sam_qos_mem_region_cfg2_reg_0-15 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0

The following image shows the lower register bit assignments.

31	16	15	0
end_addr		Reserved	

Figure 3-644 por_rnsam_sam_qos_mem_region_cfg2_reg_0-15 (low)

The following table shows the sam_qos_mem_region_cfg2_reg_0-15 lower register bit assignments.

Table 3-664 por_rnsam_sam_qos_mem_region_cfg2_reg_0-15 (low)

Bits	Field name	Description	Type	Reset
31:16	end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0
15:0	Reserved	Reserved	RO	-

sam_scg_0-511

This register repeats 511 times. It parametrized by the index from 0 to 511. Configures the Prefetch nonhash memory region #'{index}

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5000 + #{8 * [0, 1, 2, .., 510, 511]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

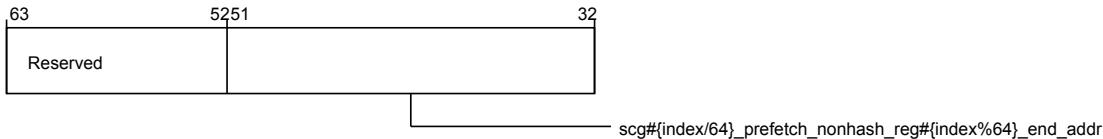


Figure 3-645 por_rnsam_sam_scg_0-511 (high)

The following table shows the sam_scg_0-511 higher register bit assignments.

Table 3-665 por_rnsam_sam_scg_0-511 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	scg#{index/64}_prefetch_nonhash_reg#{index%64}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0

The following image shows the lower register bit assignments.



Figure 3-646 por_rnsam_sam_scg_0-511 (low)

The following table shows the sam_scg_0-511 lower register bit assignments.

Table 3-666 por_rnsam_sam_scg_0-511 (low)

Bits	Field name	Description	Type	Reset
31:16	scg#{index/64}_prefetch_nonhash_reg#{index%64}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0
15:11	Reserved	Reserved	RO	-
10:0	scg#{index/64}_prefetch_nonhash_reg#{index%64}_tgtid	SN TgtID for the non-hashed region	RW	11'h0

sam_scg_0-511

This register repeats 511 times. It parametrized by the index from 0 to 511. Configures the Prefetch nonhash memory region #{index}

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h5000 + # {8 * [0, 1, 2, .., 510, 511]}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

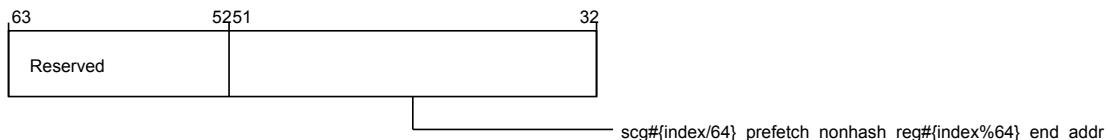


Figure 3-647 por_rnsam_sam_scg_0-511 (high)

The following table shows the sam_scg_0-511 higher register bit assignments.

Table 3-667 por_rnsam_sam_scg_0-511 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	scg#{index/64}_prefetch_nonhash_reg#{index%64}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0

The following image shows the lower register bit assignments.



Figure 3-648 por_rnsam_sam_scg_0-511 (low)

The following table shows the sam_scg_0-511 lower register bit assignments.

Table 3-668 por_rnsam_sam_scg_0-511 (low)

Bits	Field name	Description	Type	Reset
31:16	scg#{index/64}_prefetch_nonhash_reg#{index%64}_end_addr	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	36'h0
15:11	Reserved	Reserved	RO	-
10:0	scg#{index/64}_prefetch_nonhash_reg#{index%64}_tgtid	SN TgtID for the non-hashed region	RW	11'h0

sam_scg_0-63

This register repeats 63 times. It parametrized by the index from 0 to 63. Configures the Prefetch hashed memory region #{index}

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h6400 + #{8 * [0, 1, 2, .., 62, 63]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

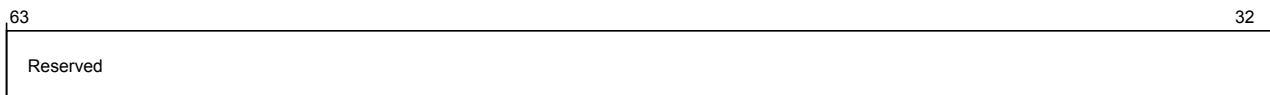


Figure 3-649 por_rnsam_sam_scg_0-63 (high)

The following table shows the sam_scg_0-63 higher register bit assignments.

Table 3-669 por_rnsam_sam_scg_0-63 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

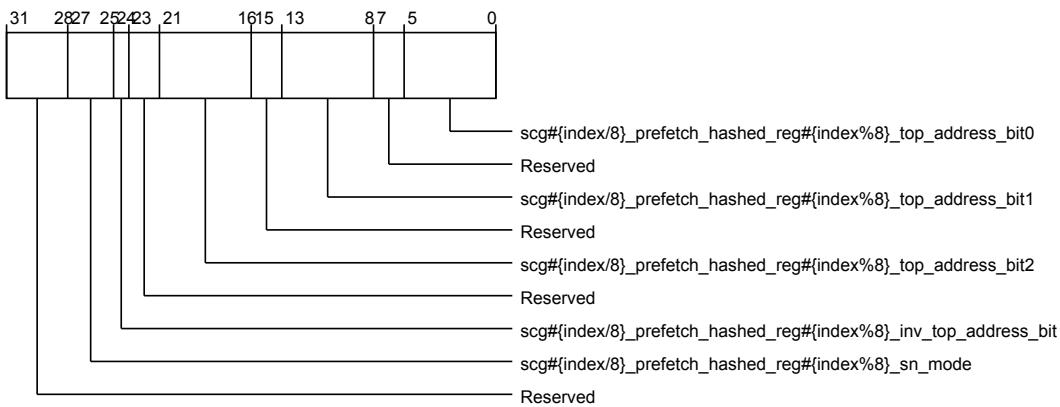


Figure 3-650 por_rnsam_sam_scg_0-63 (low)

The following table shows the sam_scg_0-63 lower register bit assignments.

Table 3-670 por_rnsam_sam_scg_0-63 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:25	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_sn_mode	SN selection mode 3'b000: Reserved 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) 3'b100: 2-SN mode (SN0, SN1) power of 2 hashing 3'b101: 4-SN mode (SN0, SN1, SN2, SN3) power of 2 hashing 3'b110: 8-SN mode (SN0, SN1, SN2, SN3, SN4, SN5, SN6, SN7) power of 2 hashing 3'b111: Reserved	RW	3'b0
24	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_inv_top_address_bit	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
23:22	Reserved	Reserved	RO	-
21:16	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_top_address_bit2	Top address bit 2	RW	6'h00
15:14	Reserved	Reserved	RO	-
13:8	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_top_address_bit1	Top address bit 1	RW	6'h00
7:6	Reserved	Reserved	RO	-
5:0	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_top_address_bit0	Top address bit 0	RW	6'h00

sam_scg_0-63

This register repeats 63 times. It parametrized by the index from 0 to 63. Configures the Prefetch hashed memory region #{index}

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h6400 + #{8 * [0, 1, 2, .., 62, 63]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

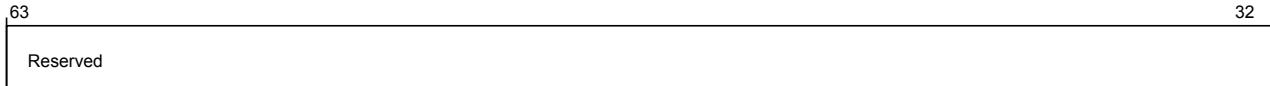


Figure 3-651 por_rnsam_sam_scg_0-63 (high)

The following table shows the sam_scg_0-63 higher register bit assignments.

Table 3-671 por_rnsam_sam_scg_0-63 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

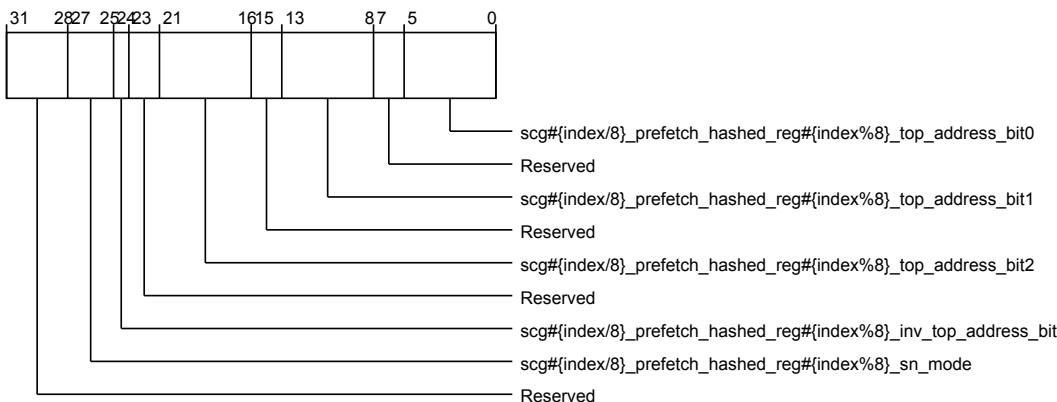


Figure 3-652 por_rnsam_sam_scg_0-63 (low)

The following table shows the sam_scg_0-63 lower register bit assignments.

Table 3-672 por_rnsam_sam_scg_0-63 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:25	scg#{index/8}_prefetch_hashed_reg#{index%8}_sn_mode	SN selection mode 3'b000: Reserved 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) 3'b100: 2-SN mode (SN0, SN1) power of 2 hashing 3'b101: 4-SN mode (SN0, SN1, SN2, SN3) power of 2 hashing 3'b110: 8-SN mode (SN0, SN1, SN2, SN3, SN4, SN5, SN6, SN7) power of 2 hashing 3'b111: Reserved	RW	3'b00

Table 3-672 por_rnsam_sam_scg_0-63 (low) (continued)

Bits	Field name	Description	Type	Reset
24	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_inv_top_address_bit	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
23:22	Reserved	Reserved	RO	-
21:16	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_top_address_bit2	Top address bit 2	RW	6'h00
15:14	Reserved	Reserved	RO	-
13:8	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_top_address_bit1	Top address bit 1	RW	6'h00
7:6	Reserved	Reserved	RO	-
5:0	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_top_address_bit0	Top address bit 0	RW	6'h00

sam_scg_0-63

This register repeats 63 times. It parametrized by the index from 0 to 63. Configures the Prefetch hashed memory region #{index}

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h6400 + #{8 * [0, 1, 2, .., 62, 63]}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.



Figure 3-653 por_rnsam_sam_scg_0-63 (high)

The following table shows the sam_scg_0-63 higher register bit assignments.

Table 3-673 por_rnsam_sam_scg_0-63 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

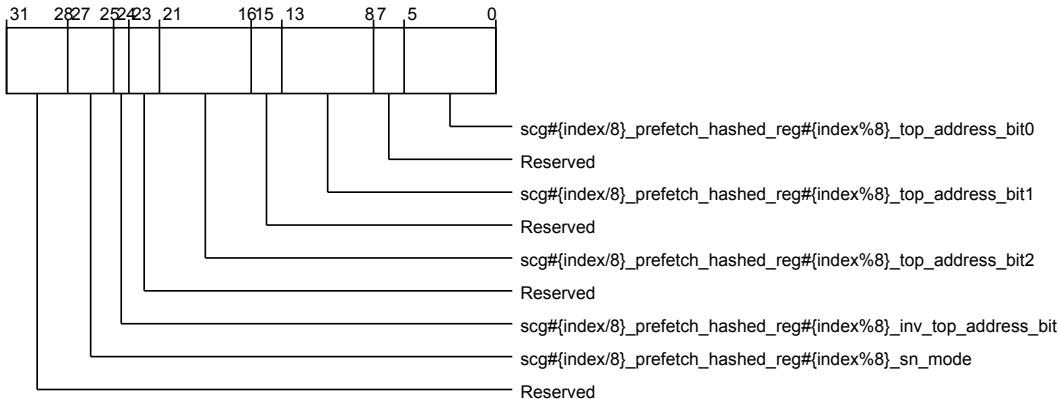


Figure 3-654 por_rnsam_sam_scg_0-63 (low)

The following table shows the sam_scg_0-63 lower register bit assignments.

Table 3-674 por_rnsam_sam_scg_0-63 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:25	scg#{index/8}_prefetch_hashed_reg#{index%8}_sn_mode	SN selection mode 3'b000: Reserved 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) 3'b100: 2-SN mode (SN0, SN1) power of 2 hashing 3'b101: 4-SN mode (SN0, SN1, SN2, SN3) power of 2 hashing 3'b110: 8-SN mode (SN0, SN1, SN2, SN3, SN4, SN5, SN6, SN7) power of 2 hashing 3'b111: Reserved	RW	3'b0
24	scg#{index/8}_prefetch_hashed_reg#{index%8}_inv_top_address_bit	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
23:22	Reserved	Reserved	RO	-
21:16	scg#{index/8}_prefetch_hashed_reg#{index%8}_top_address_bit2	Top address bit 2	RW	6'h00
15:14	Reserved	Reserved	RO	-
13:8	scg#{index/8}_prefetch_hashed_reg#{index%8}_top_address_bit1	Top address bit 1	RW	6'h00

Table 3-674 por_rnsam_sam_scg_0-63 (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-
5:0	scg#{index/ 8}_prefetch_hashed_reg#{index %8}_top_address_bit0	Top address bit 0	RW	6'h00

sys_cache_grp_sn_nodeid_reg_0-31

This register repeats 31 times. It parametrized by the index from 0 to 31. Configures hashed node IDs for system cache groups. Controls target SN node IDs #{{index}*4} to #{{index}*4 + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1000 + #{{8 * [0, 1, 2, .., 30, 31]}}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

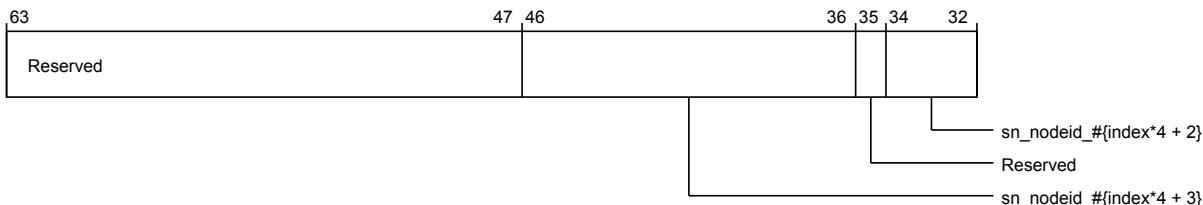


Figure 3-655 por_rnsam_sys_cache_grp_sn_nodeid_reg_0-31 (high)

The following table shows the sys_cache_grp_sn_nodeid_reg_0-31 higher register bit assignments.

Table 3-675 por_rnsam_sys_cache_grp_sn_nodeid_reg_0-31 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	sn_nodeid_#{index*4 + 3}	Default Hashed target SN node ID #{{index}*4 + 3}	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	sn_nodeid_#{index*4 + 2}	Default Hashed target SN node ID #{{index}*4 + 2}	RW	11'b000000000000

The following image shows the lower register bit assignments.

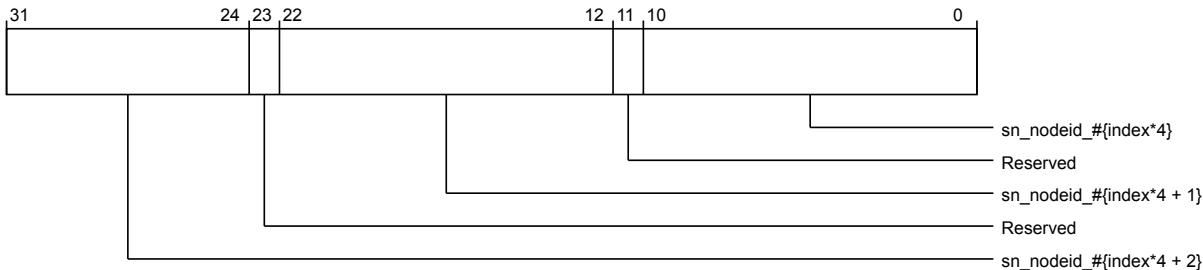


Figure 3-656 por_rnsam_sys_cache_grp_sn_nodeid_reg_0-31 (low)

The following table shows the sys_cache_grp_sn_nodeid_reg_0-31 lower register bit assignments.

Table 3-676 por_rnsam_sys_cache_grp_sn_nodeid_reg_0-31 (low)

Bits	Field name	Description	Type	Reset
31:24	sn_nodeid_{index*4 + 2}	Default Hashed target SN node ID #{index*4 + 2}	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	sn_nodeid_{index*4 + 1}	Default Hashed target SN node ID #{index*4 + 1}	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	sn_nodeid_{index*4}	Default Hashed target SN node ID #{index*4}	RW	11'b000000000000

sys_cache_grp_region_0-63

This register repeats 63 times. It parametrized by the index from 0 to 63. Configures node IDs for SCG's Default hashed Region memory. Controls target SN node IDs #{index*4} to #{index*4+3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1400 + #{8 * [0, 1, 2, .., 62, 63]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

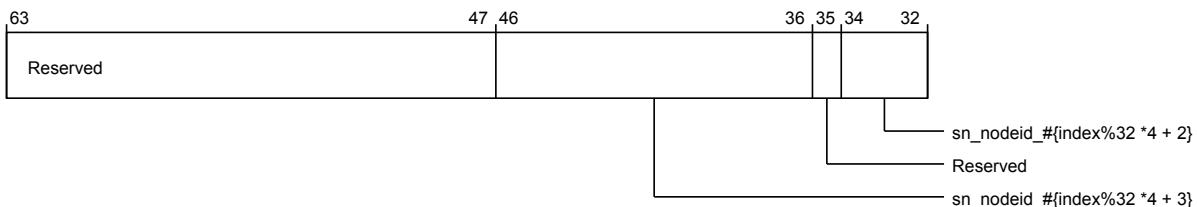


Figure 3-657 por_rnsam_sys_cache_grp_region_0-63 (high)

The following table shows the sys_cache_grp_region_0-63 higher register bit assignments.

Table 3-677 por_rnsam_sys_cache_grp_region_0-63 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	sn_nodeid_{index%32 *4 + 3}	Hashed target SN node ID #{index%32 * 4 +3}	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	sn_nodeid_{index%32 *4 + 2}	Hashed target SN node ID #{index%32 * 4 +2}	RW	11'b000000000000

The following image shows the lower register bit assignments.

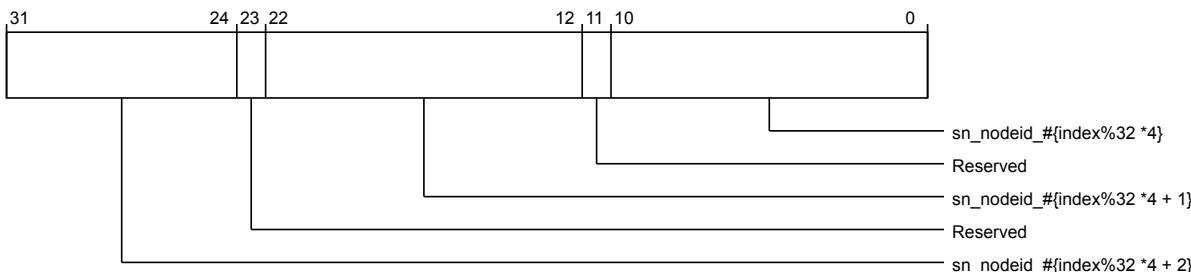


Figure 3-658 por_rnsam_sys_cache_grp_region_0-63 (low)

The following table shows the sys_cache_grp_region_0-63 lower register bit assignments.

Table 3-678 por_rnsam_sys_cache_grp_region_0-63 (low)

Bits	Field name	Description	Type	Reset
31:24	sn_nodeid_{index%32 *4 + 2}	Hashed target SN node ID #{index%32 * 4 +2}	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	sn_nodeid_{index%32 *4 + 1}	Hashed target SN node ID #{index%32 * 4 +1}	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	sn_nodeid_{index%32 *4}	Hashed target SN node ID #{index%32 * 4}	RW	11'b000000000000

sys_cache_grp_hashed_regions_sn_nodeid_reg_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Configures SN node IDs for SCG's Hashed groups in the HNSAM . Controls target SN node IDs #{index*4} to #{index*4 + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h6600 + # {8 * [0, 1, 2, .., 14, 15]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

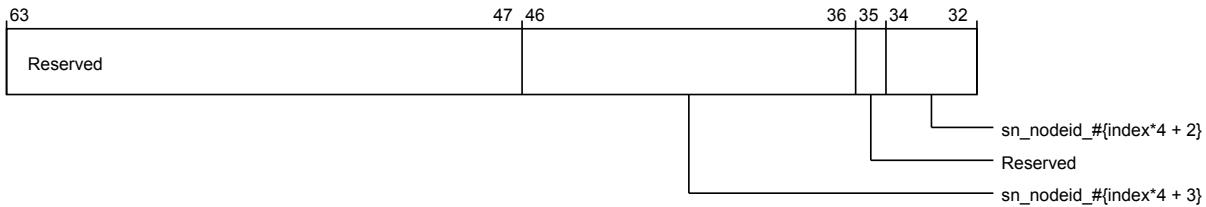


Figure 3-659 por_rnsam_sys_cache_grp_hashed_regions_sn_nodeid_reg_0-15 (high)

The following table shows the sys_cache_grp_hashed_regions_sn_nodeid_reg_0-15 higher register bit assignments.

Table 3-679 por_rnsam_sys_cache_grp_hashed_regions_sn_nodeid_reg_0-15 (high)

Bits	Field name	Description	Type	Reset
63:47	Reserved	Reserved	RO	-
46:36	sn_nodeid_{index*4 + 3}	Hashed target SN node ID #{index*4 + 3}	RW	11'b000000000000
35	Reserved	Reserved	RO	-
34:32	sn_nodeid_{index*4 + 2}	Hashed target SN node ID #{index*4 + 2}	RW	11'b000000000000

The following image shows the lower register bit assignments.

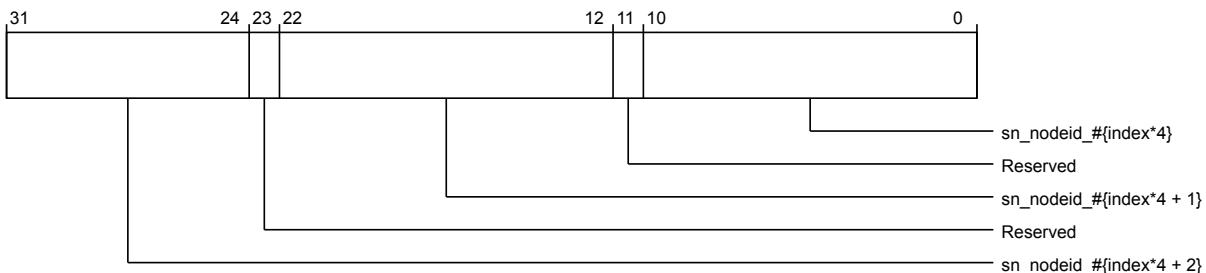


Figure 3-660 por_rnsam_sys_cache_grp_hashed_regions_sn_nodeid_reg_0-15 (low)

The following table shows the sys_cache_grp_hashed_regions_sn_nodeid_reg_0-15 lower register bit assignments.

Table 3-680 por_rnsam_sys_cache_grp_hashed_regions_sn_nodeid_reg_0-15 (low)

Bits	Field name	Description	Type	Reset
31:24	sn_nodeid_{index*4 + 2}	Hashed target SN node ID #{index*4 + 2}	RW	11'b000000000000
23	Reserved	Reserved	RO	-
22:12	sn_nodeid_{index*4 + 1}	Hashed target SN node ID #{index*4 + 1}	RW	11'b000000000000
11	Reserved	Reserved	RO	-
10:0	sn_nodeid_{index*4}	Hashed target SN node ID #{index*4}	RW	11'b000000000000

rnsam_status

Functions as the default and programming mode status register.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1100
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

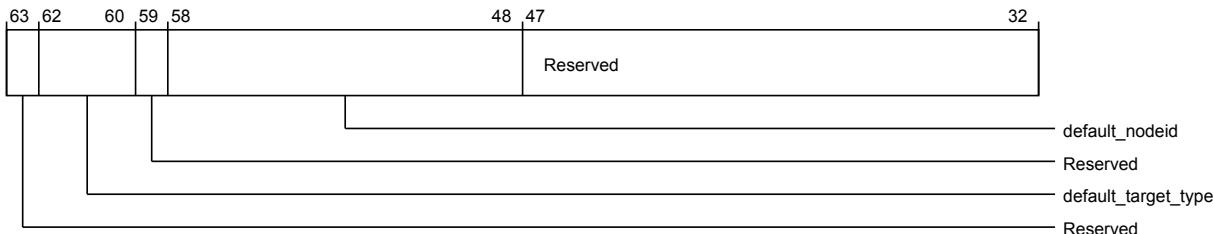


Figure 3-661 por_rnsam_rnsam_status (high)

The following table shows the rnsam_status higher register bit assignments.

Table 3-681 por_rnsam_rnsam_status (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:60	default_target_type	Indicates node type 3'b000: HN-F 3'b001: HN-I 3'b010: CXRA 3'b011: HN-P 3'b100: PCI-CXRA Others: Reserved CONSTRAINT: Only applicable for RN-I	RW	3'b001
59	Reserved	Reserved	RO	-
58:48	default_nodeid	Default Node ID	RW	Configuration dependent
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

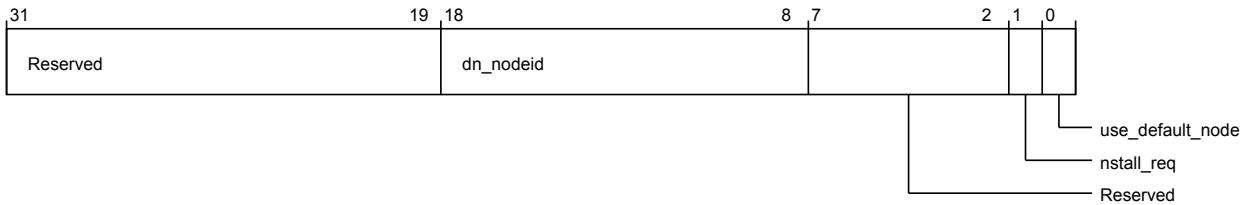


Figure 3-662 por_rnsam_rnsam_status (low)

The following table shows the rnsam_status lower register bit assignments.

Table 3-682 por_rnsam_rnsam_status (low)

Bits	Field name	Description	Type	Reset
31:19	Reserved	Reserved	RO	-
18:8	dn_nodeid	DN Node ID for DN operations	RW	11'b0
7:2	Reserved	Reserved	RO	-
1	nstall_req	Indicates RN SAM is programmed and ready 1'b0: STALL requests 1'b1: UNSTALL requests	RW	1'b0
0	use_default_node	Indicates target ID selection mode 1'b0: Enables RN SAM to hash address bits and generate target ID 1'b1: Uses default target ID	RW	1'b1

gic_mem_region_reg

Configures GIC memory region.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1108

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

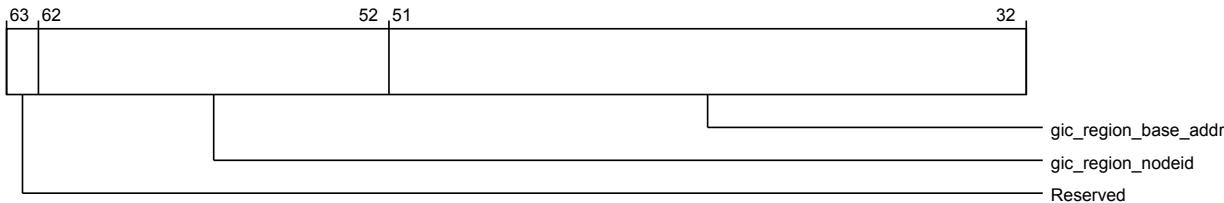


Figure 3-663 por_rnsam_gic_mem_region_reg (high)

The following table shows the gic_mem_region_reg higher register bit assignments.

Table 3-683 por_rnsam_gic_mem_region_reg (high)

Bits	Field name	Description	Type	Reset
63	Reserved	Reserved	RO	-
62:52	gic_region_nodeid	GIC node ID	RW	11'b000000000000
51:32	gic_region_base_addr	Base address of the GIC memory region CONSTRAINT: Must be an integer multiple of region size	RW	36'h000000000

The following image shows the lower register bit assignments.

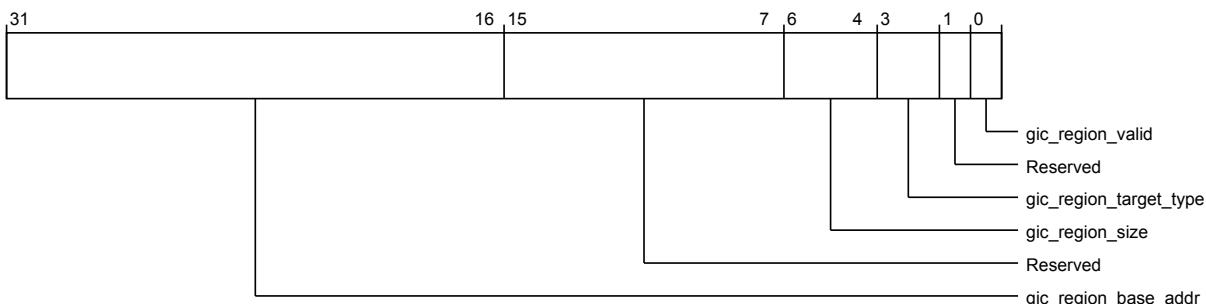


Figure 3-664 por_rnsam_gic_mem_region_reg (low)

The following table shows the gic_mem_region_reg lower register bit assignments.

Table 3-684 por_rnsam_gic_mem_region_reg (low)

Bits	Field name	Description	Type	Reset
31:16	gic_region_base_addr	Base address of the GIC memory region CONSTRAINT: Must be an integer multiple of region size	RW	36'h000000000
15:7	Reserved	Reserved	RO	-
6:4	gic_region_size	GIC memory region size 3'b000: 64KB 3'b001: 128KB 3'b010: 256KB 3'b011: 512KB CONSTRAINT: Memory region must be a power of 2.	RW	3'b000

Table 3-684 por_rnsam_gic_mem_region_reg (low) (continued)

Bits	Field name	Description	Type	Reset
3:2	gic_region_target_type	Indicates node type 2'b00: HN-F 2'b01: HN-I 2'b10: CXRA 2'b11: HN-P CONSTRAINT: Only applicable for RN-I	RW	2'b00
1	Reserved	Reserved	RO	-
0	gic_region_valid	Memory region 1 valid 1'b0: Not valid 1'b1: Valid for memory region comparison	RW	1'b0

sam_generic_regs_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Configuration register for the custom logic

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1600 + #{8 * [0, 1, 2, 3, 4, 5, 6, 7]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rnsam_secure_register_groups_override.mem_range

The following image shows the higher register bit assignments.

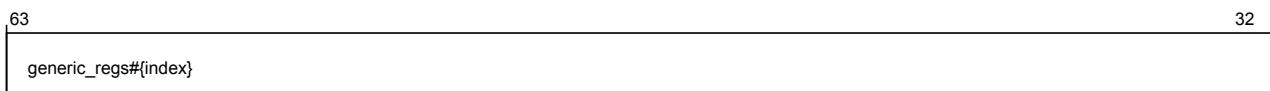


Figure 3-665 por_rnsam_sam_generic_regs_0-7 (high)

The following table shows the sam_generic_regs_0-7 higher register bit assignments.

Table 3-685 por_rnsam_sam_generic_regs_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	generic_regs#{index}	Configuration register for the custom logic	RW	64'h0

The following image shows the lower register bit assignments.



Figure 3-666 por_rnsam_sam_generic_regs_0-7 (low)

The following table shows the sam_generic_regs_0-7 lower register bit assignments.

Table 3-686 por_rnsam_sam_generic_regs_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	generic_regs#{index}	Configuration register for the custom logic	RW	64'h0

3.3.11 HN-F MPAM_S register descriptions

This section lists the HN-F MPAM_S registers.

por_hnf_mpam_s_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

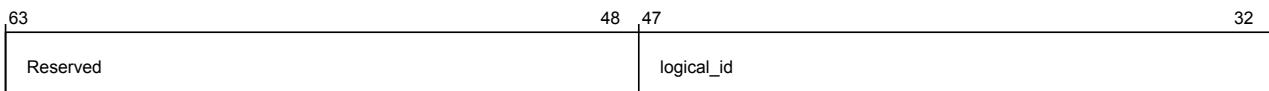


Figure 3-667 por_hnf_mpam_s_por_hnf_mpam_s_node_info (high)

The following table shows the por_hnf_mpam_s_node_info higher register bit assignments.

Table 3-687 por_hnf_mpam_s_por_hnf_mpam_s_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

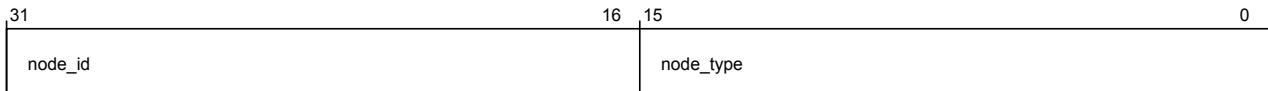


Figure 3-668 por_hnf_mpam_s_por_hnf_mpam_s_node_info (low)

The following table shows the por_hnf_mpam_s_node_info lower register bit assignments.

Table 3-688 por_hnf_mpam_s_por_hnf_mpam_s_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0008

por_hnf_mpam_s_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

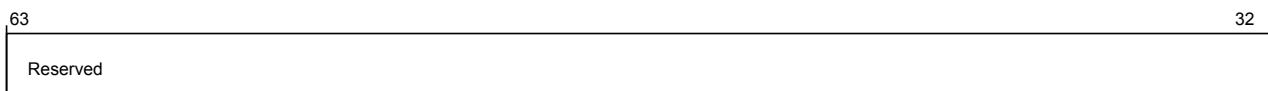


Figure 3-669 por_hnf_mpam_s_por_hnf_mpam_s_child_info (high)

The following table shows the por_hnf_mpam_s_child_info higher register bit assignments.

Table 3-689 por_hnf_mpam_s_por_hnf_mpam_s_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

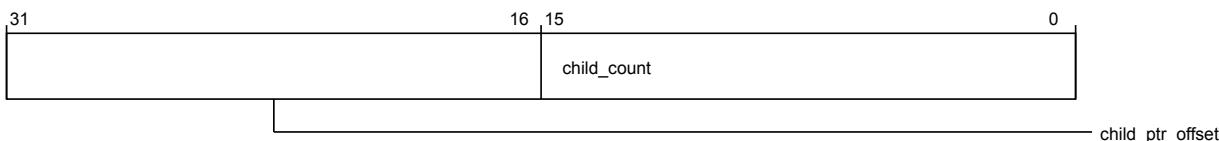


Figure 3-670 por_hnf_mpam_s_por_hnf_mpam_s_child_info (low)

The following table shows the por_hnf_mpam_s_child_info lower register bit assignments.

Table 3-690 por_hnf_mpam_s_por_hnf_mpam_s_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_hnf_mpam_s_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-671 por_hnf_mpam_s_por_hnf_mpam_s_secure_register_groups_override (high)

The following table shows the por_hnf_mpam_s_secure_register_groups_override higher register bit assignments.

Table 3-691 por_hnf_mpam_s_por_hnf_mpam_s_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

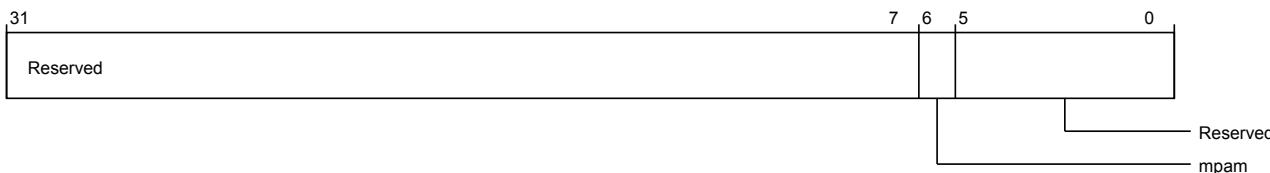


Figure 3-672 por_hnf_mpam_s_por_hnf_mpam_s_secure_register_groups_override (low)

The following table shows the por_hnf_mpam_s_secure_register_groups_override lower register bit assignments.

Table 3-692 por_hnf_mpam_s_por_hnf_mpam_s_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:7	Reserved	Reserved	RO	-
6	mpam	Allows non-secure access to secure MPAM registers	RW	1'b0
5:0	Reserved	Reserved	RO	-

por_hnf_mpam_sidr

MPAM features Secure ID register. This is Secure (S) register only.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1008

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-673 por_hnf_mpam_s_por_hnf_mpam_sidr (high)

The following table shows the por_hnf_mpam_sidr higher register bit assignments.

Table 3-693 por_hnf_mpam_s_por_hnf_mpam_sidr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

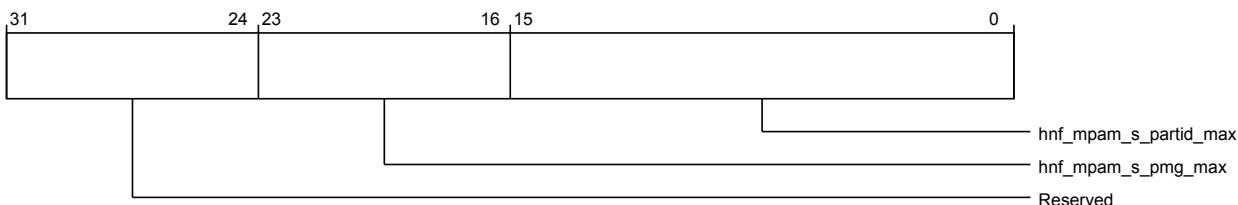


Figure 3-674 por_hnf_mpam_s_por_hnf_mpam_sidr (low)

The following table shows the por_hnf_mpam_sidr lower register bit assignments.

Table 3-694 por_hnf_mpam_s_por_hnf_mpam_sidr (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_mpam_s_pmg_max	Maximum value of secure PMG supported by this HN-F	RO	Configuration dependent
15:0	hnf_mpam_s_partid_max	Maximum value of secure PARTID supported by this HN-F	RO	Configuration dependent

por_hnf_s_mpam_ecr

MPAM Error Control Register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h10F0

Register reset 64'b0

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

Secure group override por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

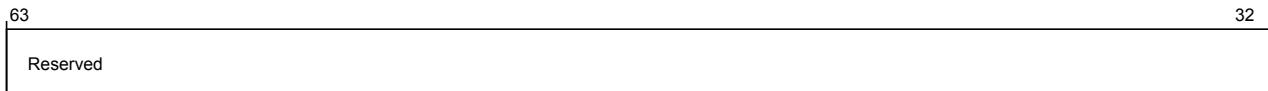


Figure 3-675 por_hnf_mpam_s_por_hnf_s_mpam_ecr (high)

The following table shows the por_hnf_s_mpam_ecr higher register bit assignments.

Table 3-695 por_hnf_mpam_s_por_hnf_s_mpam_ecr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

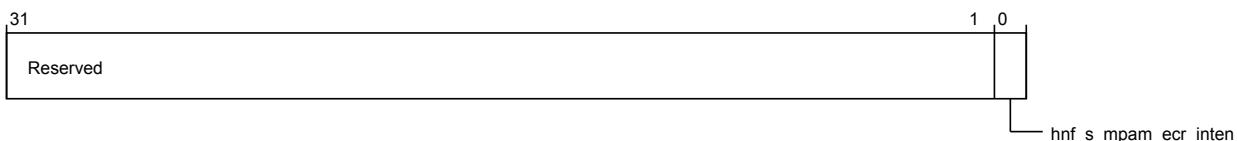


Figure 3-676 por_hnf_mpam_s_por_hnf_s_mpam_ecr (low)

The following table shows the por_hnf_s_mpam_ecr lower register bit assignments.

Table 3-696 por_hnf_mpam_s_por_hnf_s_mpam_ecr (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	hnf_s_mpam_ecr_inten	Interrupt Enable. When INTEN = 0, MPAM error interrupts are not generated. When INTEN = 1, MPAM error interrupts are generated.	RW	1'h0

por_hnf_s_mpam_esr

MPAM Error Status Register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h10F8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.



Figure 3-677 por_hnf_mpam_s_por_hnf_s_mpam_esr (high)

The following table shows the por_hnf_s_mpam_esr higher register bit assignments.

Table 3-697 por_hnf_mpam_s_por_hnf_s_mpam_esr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

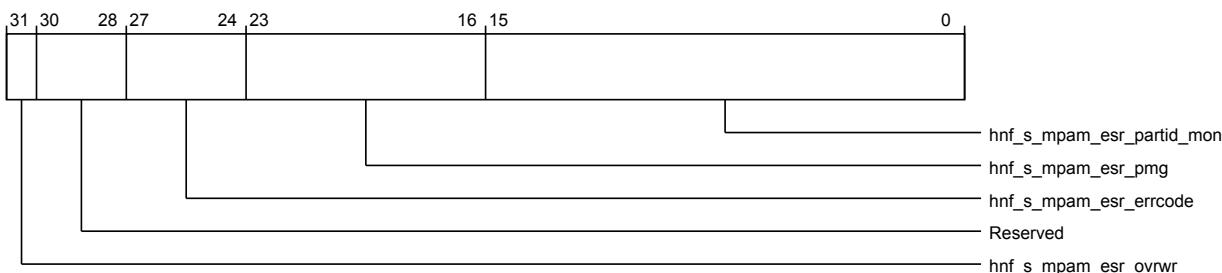


Figure 3-678 por_hnf_mpam_s_por_hnf_s_mpam_esr (low)

The following table shows the por_hnf_s_mpam_esr lower register bit assignments.

Table 3-698 por_hnf_mpam_s_por_hnf_s_mpam_esr (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_mpam_esr_ovrwr	Overwritten. If 0 and ERRCODE is zero, no errors have occurred. If 0 and ERRCODE is non-zero, a single error has occurred and is recorded in this register. If 1 and ERRCODE is non-zero, multiple errors have occurred and this register records the most recent error. The state where this bit is 1 and ERRCODE is zero is not produced by hardware and is only reached when software writes this combination into this register.	RW	1'h0
30:28	Reserved	Reserved	RO	-
27:24	hnf_s_mpam_esr_errcode	Error code	RW	4'h0
23:16	hnf_s_mpam_esr_pmg	PMG captured if the error code captures PMG, otherwise 0x0000.	RW	8'h0
15:0	hnf_s_mpam_esr_partid_mon	PARTID captured if the error code captures PARTID. MON selector captured if the error code captures MON. Otherwise 0x0000.	RW	16'h0

por_hnf_s_mpamcfg_part_sel

MPAM partition configuration selection register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1100
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

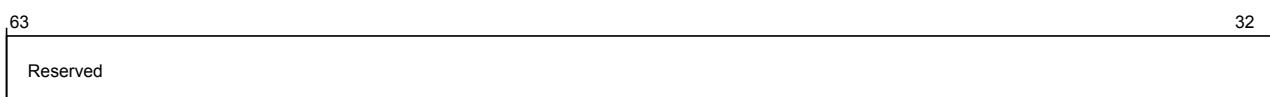


Figure 3-679 por_hnf_mpam_s_por_hnf_s_mpamcfg_part_sel (high)

The following table shows the por_hnf_mpamcfg_part_sel higher register bit assignments.

Table 3-699 por_hnf_mpam_s_por_hnf_s_mpamcfg_part_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

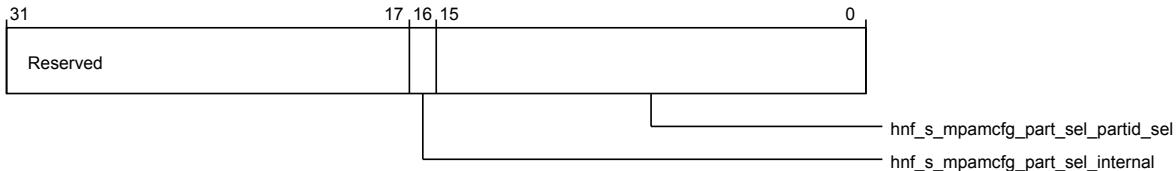


Figure 3-680 por_hnf_mpam_s_por_hnf_s_mpamcfg_part_sel (low)

The following table shows the por_hnf_s_mpamcfg_part_sel lower register bit assignments.

Table 3-700 por_hnf_mpam_s_por_hnf_s_mpamcfg_part_sel (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	hnf_s_mpamcfg_part_sel_internal	If MPAMF_IDR.HAS_PARTID_NRW = 0, this field is RAZ/WI. If MPAMF_IDR.HAS_PARTID_NRW = 1, this bit decides how to interprete PARTID_SEL.	RW	1'h0
15:0	hnf_s_mpamcfg_part_sel_partid_sel	Selects the partition ID to configure.	RW	16'h0

por_hnf_s_mpamcfg_cmax

MPAM cache maximum capacity partition configuration register. This register is a banked seperately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1108
Register reset	64'b111111
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

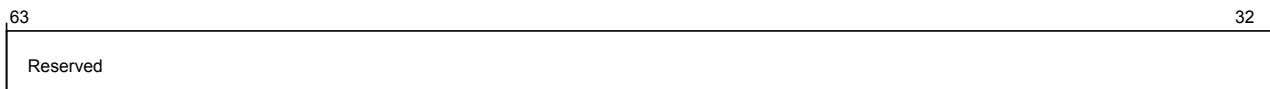


Figure 3-681 por_hnf_mpam_s_por_hnf_s_mpamcfg_cmax (high)

The following table shows the por_hnf_s_mpamcfg_cmax higher register bit assignments.

Table 3-701 por_hnf_mpam_s_por_hnf_s_mpamcfg_cmax (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

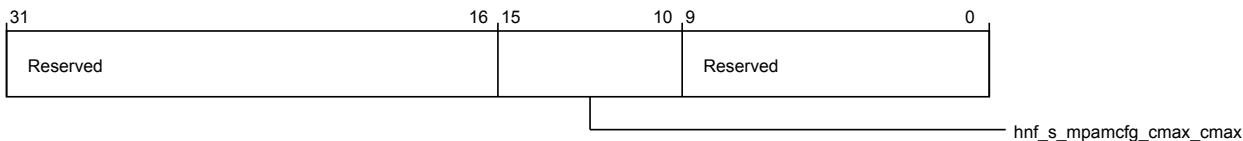


Figure 3-682 por_hnf_mpam_s_por_hnf_s_mpamcfg_cmax (low)

The following table shows the por_hnf_s_mpamcfg_cmax lower register bit assignments.

Table 3-702 por_hnf_mpam_s_por_hnf_s_mpamcfg_cmax (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:10	hnf_s_mpamcfg_cmax_cmax	Maximum cache capacity usage in fixed-point fraction of the cache capacity by the partition selected by MPAMCFG_PART_SEL.	RW	6'h3f
9:0	Reserved	Reserved	RO	-

por_hnf_s_mpamcfg_mbw_min

MPAM memory minimum bandwidth partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1200
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

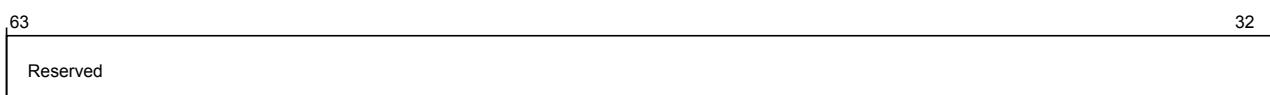


Figure 3-683 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_min (high)

The following table shows the por_hnf_s_mpamcfg_mbw_min higher register bit assignments.

Table 3-703 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_min (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

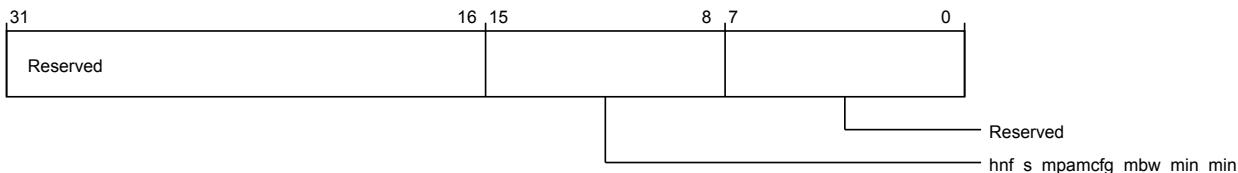


Figure 3-684 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_min (low)

The following table shows the por_hnf_s_mpamcfg_mbw_min lower register bit assignments.

Table 3-704 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_min (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:8	hnf_s_mpamcfg_mbw_min_min	Memory minimum bandwidth allocated to the partition selected by MPAMCFG_PART_SEL.	RW	8'h0
7:0	Reserved	Reserved	RO	-

por_hnf_s_mpamcfg_mbw_max

MPAM memory maximum bandwidth partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1208
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

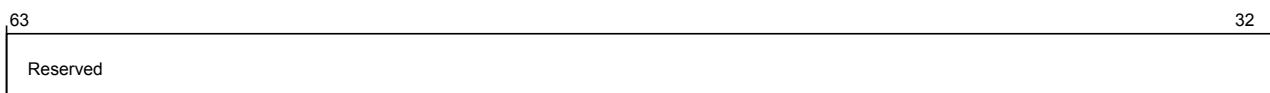


Figure 3-685 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_max (high)

The following table shows the por_hnf_s_mpamcfg_mbw_max higher register bit assignments.

Table 3-705 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_max (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

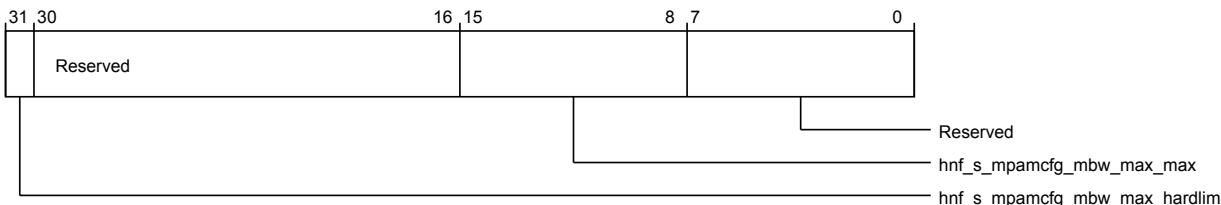


Figure 3-686 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_max (low)

The following table shows the por_hnf_s_mpamcfg_mbw_max lower register bit assignments.

Table 3-706 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_max (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_mpamcfg_mbw_max_hardlim	0: When MAX bandwidth is exceeded, the partition may contend with a low preference for downstream bandwidth beyond its maximum bandwidth. 1: When MAX bandwidth is exceeded, the partition may not be use any more bandwidth until its memory bandwidth measurement falls below the maximum limit.	RW	1'h0
30:16	Reserved	Reserved	RO	-
15:8	hnf_s_mpamcfg_mbw_max_max	Memory maximum bandwidth allocated to the partition selected by MPAMCFG_PART_SEL.	RW	8'h0
7:0	Reserved	Reserved	RO	-

por_hnf_s_mpamcfg_mbw_winwd

MPAM memory bandwidth partitioning window width register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1220
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

63	Reserved	32
----	----------	----

Figure 3-687 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_winwd (high)

The following table shows the por_hnf_s_mpamcfg_mbw_winwd higher register bit assignments.

Table 3-707 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_winwd (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

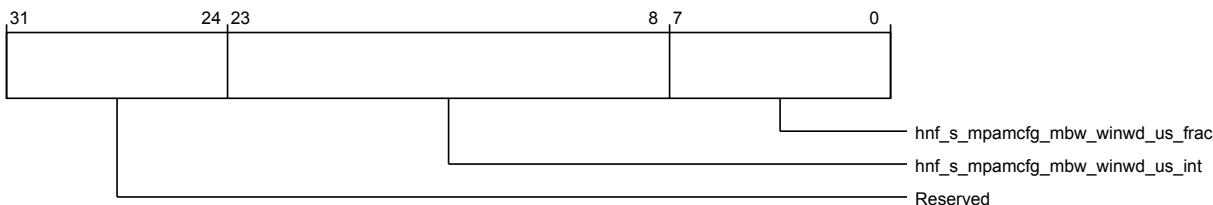


Figure 3-688 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_winwd (low)

The following table shows the por_hnf_s_mpamcfg_mbw_winwd lower register bit assignments.

Table 3-708 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_winwd (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:8	hnf_s_mpamcfg_mbw_winwd_us_int	Memory bandwidth accounting period integer microseconds.	RW	16'h0
7:0	hnf_s_mpamcfg_mbw_winwd_us_frac	Memory bandwidth accounting period fractions of a microsecond.	RW	8'h0

por_hnf_s_mpamcfg_pri

MPAM priority partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1400

Register reset 64'b0

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

Secure group override por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

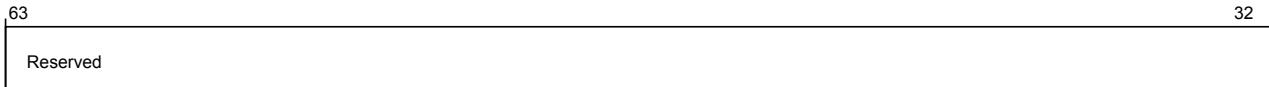


Figure 3-689 por_hnf_mpam_s_por_hnf_s_mpamcfg_pri (high)

The following table shows the por_hnf_s_mpamcfg_pri higher register bit assignments.

Table 3-709 por_hnf_mpam_s_por_hnf_s_mpamcfg_pri (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

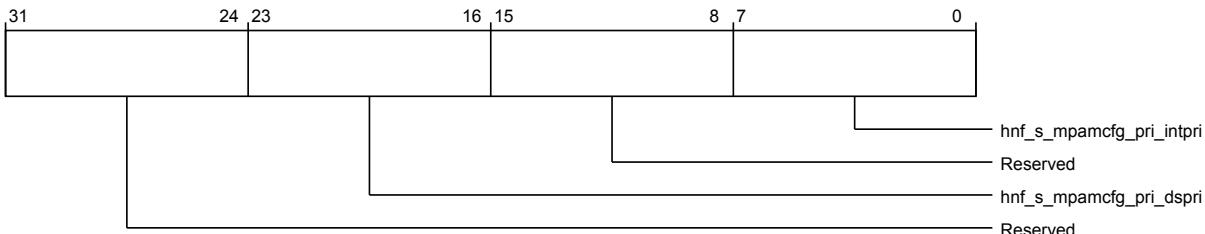


Figure 3-690 por_hnf_mpam_s_por_hnf_s_mpamcfg_pri (low)

The following table shows the por_hnf_s_mpamcfg_pri lower register bit assignments.

Table 3-710 por_hnf_mpam_s_por_hnf_s_mpamcfg_pri (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_s_mpamcfg_pri_dspri	If HAS_DSPRI is 1, this field is a priority value applied to downstream communications from this memory system component for transactions of the partition selected by MPAMCFG_PART_SEL.	RW	8'h0
15:8	Reserved	Reserved	RO	-
7:0	hnf_s_mpamcfg_pri_intpri	If HAS_INTPRI is 1, this field is a priority value applied internally inside this memory system component for transactions of the partition selected by MPAMCFG_PART_SEL.	RW	8'h0

por_hnf_s_mpamcfg_mbw_prop

Memory bandwidth proportional stride partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1500
Register reset	64'b0

Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

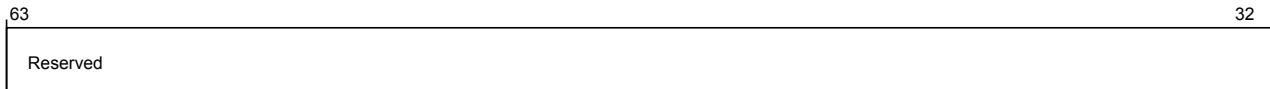


Figure 3-691 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_prop (high)

The following table shows the por_hnf_s_mpamcfg_mbw_prop higher register bit assignments.

Table 3-711 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_prop (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

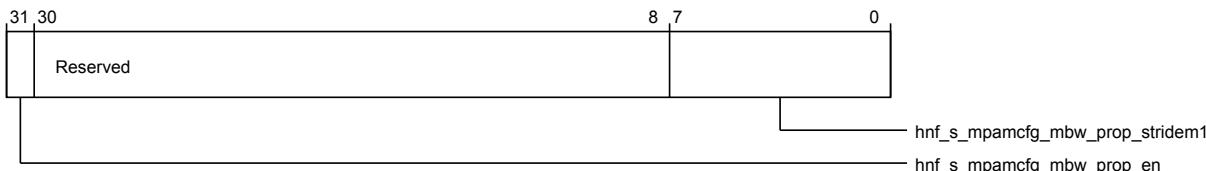


Figure 3-692 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_prop (low)

The following table shows the por_hnf_s_mpamcfg_mbw_prop lower register bit assignments.

Table 3-712 por_hnf_mpam_s_por_hnf_s_mpamcfg_mbw_prop (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_mpamcfg_mbw_prop_en	0: The selected partition is not regulated by proportional stride bandwidth partitioning. 1: The selected partition has bandwidth usage regulated by proportional stride bandwidth partitioning as controlled by STRIDEM1.	RW	1'h0
30:8	Reserved	Reserved	RO	-
7:0	hnf_s_mpamcfg_mbw_prop_stridem1	Normalized cost of a bandwidth consumption by the partition. STRIDEM1 is the stride for the partition minus one.	RW	8'h0

por_hnf_s_mpamcfg_intpartid

MPAM internal partition narrowing configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
-------------	----

Register width (Bits)	64
Address offset	16'h1600
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.



Figure 3-693 por_hnf_mpam_s_por_hnf_s_mpamcfg_intpartid (high)

The following table shows the por_hnf_s_mpamcfg_intpartid higher register bit assignments.

Table 3-713 por_hnf_mpam_s_por_hnf_s_mpamcfg_intpartid (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

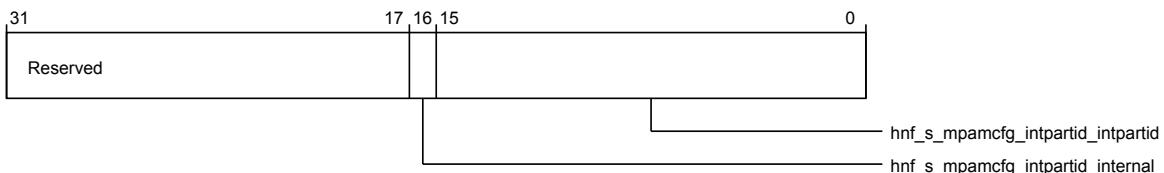


Figure 3-694 por_hnf_mpam_s_por_hnf_s_mpamcfg_intpartid (low)

The following table shows the por_hnf_s_mpamcfg_intpartid lower register bit assignments.

Table 3-714 por_hnf_mpam_s_por_hnf_s_mpamcfg_intpartid (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	hnf_s_mpamcfg_intpartid_internal	This bit must be 1 when written to the register. If written as 0, the write will not update the reqPARTID to intPARTID association.	RW	1'h0
15:0	hnf_s_mpamcfg_intpartid_intpartid	This field contains the intPARTID mapped to the reqPARTID in MPAMCFG_PART_SEL.	RW	16'h0

por_hnf_s_msmon_cfg_mon_sel

Memory system performance monitor selection register. This register is a banked separately for S and NS.

Its characteristics are:

Type	RW
-------------	----

Register width (Bits)	64
Address offset	16'h1800
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.



Figure 3-695 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mon_sel (high)

The following table shows the por_hnf_s_msmon_cfg_mon_sel higher register bit assignments.

Table 3-715 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mon_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-696 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mon_sel (low)

The following table shows the por_hnf_s_msmon_cfg_mon_sel lower register bit assignments.

Table 3-716 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mon_sel (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	hnf_s_msmon_cfg_mon_sel_mon_sel	Selects the performance monitor to configure.	RW	16'h0

por_hnf_s_msmon_capt_evt

Memory system performance monitoring capture event generation register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1808

Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

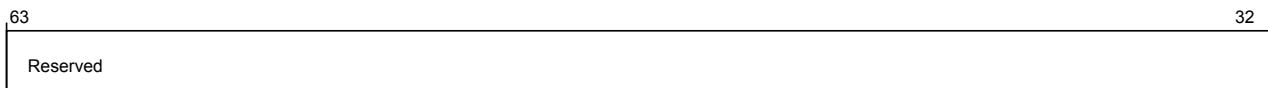


Figure 3-697 por_hnf_mpam_s_por_hnf_s_msmon_capt_evnt (high)

The following table shows the por_hnf_s_msmon_capt_evnt higher register bit assignments.

Table 3-717 por_hnf_mpam_s_por_hnf_s_msmon_capt_evnt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

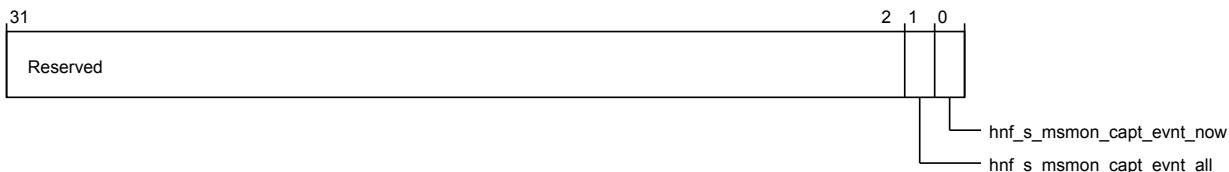


Figure 3-698 por_hnf_mpam_s_por_hnf_s_msmon_capt_evnt (low)

The following table shows the por_hnf_s_msmon_capt_evnt lower register bit assignments.

Table 3-718 por_hnf_mpam_s_por_hnf_s_msmon_capt_evnt (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1	hnf_s_msmon_capt_evnt_all	In secure version, if ALL written as 1 and NOW is also written as 1, signal a capture event to secure and non-secure monitors in this memory system component with CAPT_EVNT = 7. If written as 0 and NOW is written as 1, signal a capture event to secure monitors in this memory system component with CAPT_EVNT = 7. In non-secure version if NOW is written as 1, signal a capture event to non-secure monitors in this memory system component with CAPT_EVNT = 7.	RW	1'h0
0	hnf_s_msmon_capt_evnt_now	When written as 1, this bit causes an event to all monitors in this memory system component with CAPT_EVNT set to the value of 7. When this bit is written as 0, no event is signalled.	RW	1'h0

por_hnf_s_msmon_cfg_csu_flt

Memory system performance monitor configure cache storage monitor filter register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1810
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.



Figure 3-699 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_flt (high)

The following table shows the por_hnf_s_msmon_cfg_csu_flt higher register bit assignments.

Table 3-719 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_flt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

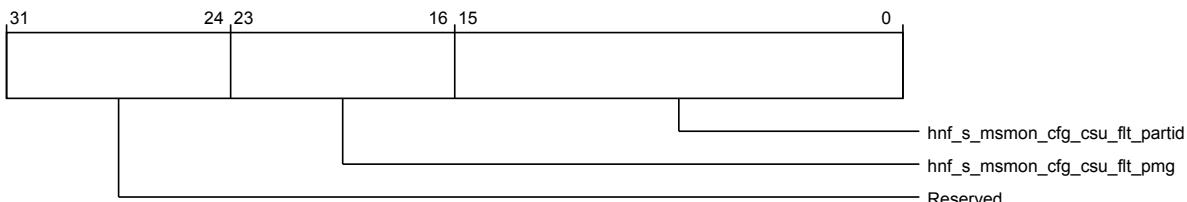


Figure 3-700 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_flt (low)

The following table shows the por_hnf_s_msmon_cfg_csu_flt lower register bit assignments.

Table 3-720 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_flt (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_s_msmon_cfg_csu_flt_pmg	Configures the cache storage usage performance monitor to a PMG. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures storage usage by cache lines labelled with both the configured PARTID and PMG.	RW	8'h0
15:0	hnf_s_msmon_cfg_csu_flt_partid	Configures the cache storage usage performance monitor to a PARTID. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures the storage usage by cache lines labelled with both the configured PARTID and PMG.	RW	16'h0

por_hnf_s_msmon_cfg_csu_ctl

Memory system performance monitor configure cache storage usage monitor control register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1818
Register reset	64'b000000000000000010011
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

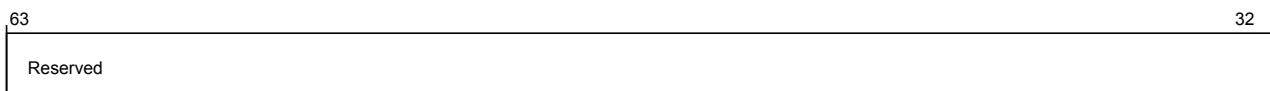


Figure 3-701 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_ctl (high)

The following table shows the por_hnf_s_msmon_cfg_csu_ctl higher register bit assignments.

Table 3-721 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

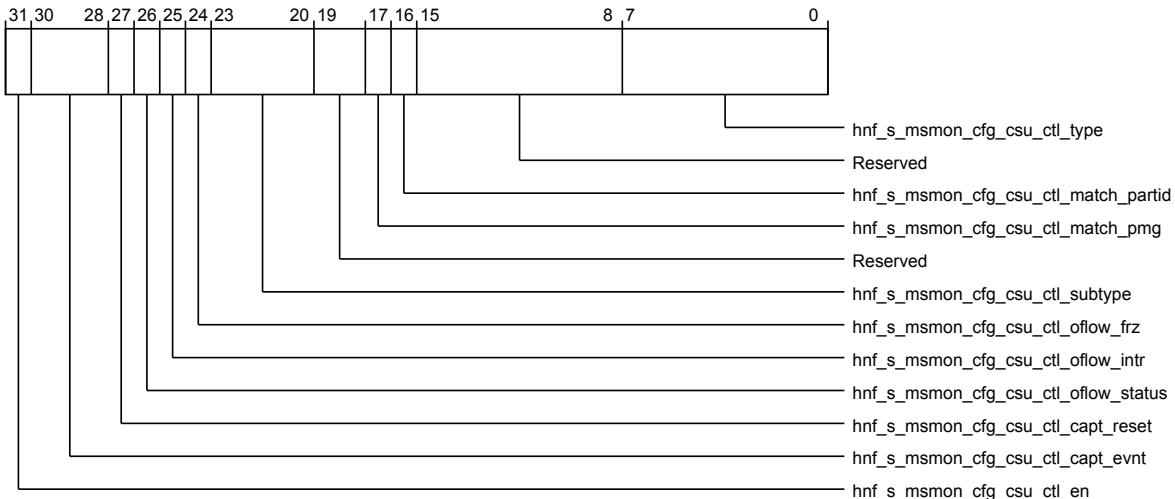


Figure 3-702 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_ctl (low)

The following table shows the por_hnf_s_msmon_cfg_csu_ctl lower register bit assignments.

Table 3-722 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_ctl (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_msmon_cfg_csu_ctl_en	0: The monitor is disabled and must not collect any information. 1: The monitor is enabled to collect information according to its configuration.	RW	1'h0
30:28	hnf_s_msmon_cfg_csu_ctl_capt_evnt	Select the event that triggers capture from the following: 0: No capture event is triggered. 1: External capture event 1 (optional but recommended)	RW	3'h0
27	hnf_s_msmon_cfg_csu_ctl_capt_reset	Capture is not implemented for the CSU monitor type.	RW	1'h0
26	hnf_s_msmon_cfg_csu_ctl_oflow_status	0: No overflow has occurred. 1: At least one overflow has occurred since this bit was last written.	RW	1'h0
25	hnf_s_msmon_cfg_csu_ctl_oflow_intr	0: No interrupt. 1: On overflow, an implementation-specific interrupt is signalled.	RW	1'h0
24	hnf_s_msmon_cfg_csu_ctl_oflow_frz	0: Monitor count wraps on overflow. 1: Monitor count freezes on overflow. The frozen value may be 0 or another value if the monitor overflowed with an increment larger than 1.	RW	1'h0
23:20	hnf_s_msmon_cfg_csu_ctl_subtype	Not currently used for CSU monitors, but reserved for future use.	RW	4'h0
19:18	Reserved	Reserved	RO	-
17	hnf_s_msmon_cfg_csu_ctl_match_pmg	0: Monitor storage used by all PMG values. 1: Only monitor storage used with the PMG value matching MSMON_CFG_CSU_FLT.PMG.	RW	1'h0

Table 3-722 por_hnf_mpam_s_por_hnf_s_msmon_cfg_csu_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
16	hnf_s_msmon_cfg_csu_ctl_match_partid	0: Monitor storage used by all PARTIDs. 1: Only monitor storage used with the PARTID matching MSMON_CFG_CSU_FLT.PARTID.	RW	1'h0
15:8	Reserved	Reserved	RO	-
7:0	hnf_s_msmon_cfg_csu_ctl_type	Read-only: Constant type indicating the type of the monitor. CSU monitor is TYPE = 0x43.	RW	8'h43

por_hnf_s_msmon_cfg_mbwu_filt

Memory system performance monitor configure memory bandwidth usage monitor filter register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1820
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.



Figure 3-703 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_filt (high)

The following table shows the por_hnf_s_msmon_cfg_mbwu_filt higher register bit assignments.

Table 3-723 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_filt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

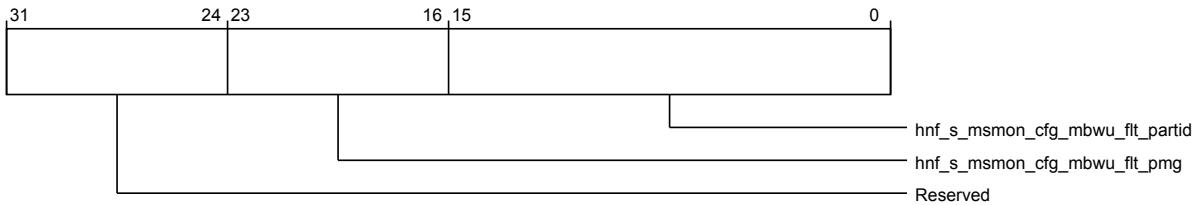


Figure 3-704 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_flt (low)

The following table shows the por_hnf_s_msmon_cfg_mbwu_flt lower register bit assignments.

Table 3-724 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_flt (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_s_msmon_cfg_mbwu_flt_pmg	Configures the memory bandwidth usage performance monitor to a PMG. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures the memory bandwidth used by requests labelled with both the configured PARTID and PMG.	RW	8'h0
15:0	hnf_s_msmon_cfg_mbwu_flt_partid	Configures the memory bandwidth usage performance monitor to a PARTID. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures the memory bandwidth used by requests labelled with both the configured PARTID and PMG.	RW	16'h0

por_hnf_s_msmon_cfg_mbwu_ctl

Memory system performance monitor configure memory bandwidth usage monitor control register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1828
Register reset	64'b000000000000000010010
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

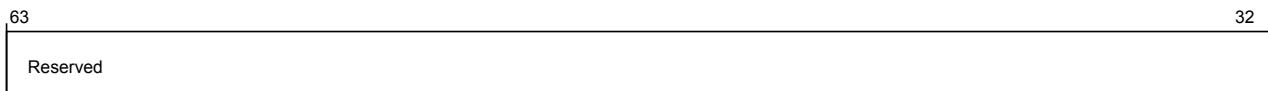


Figure 3-705 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_ctl (high)

The following table shows the por_hnf_s_msmon_cfg_mbwu_ctl higher register bit assignments.

Table 3-725 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

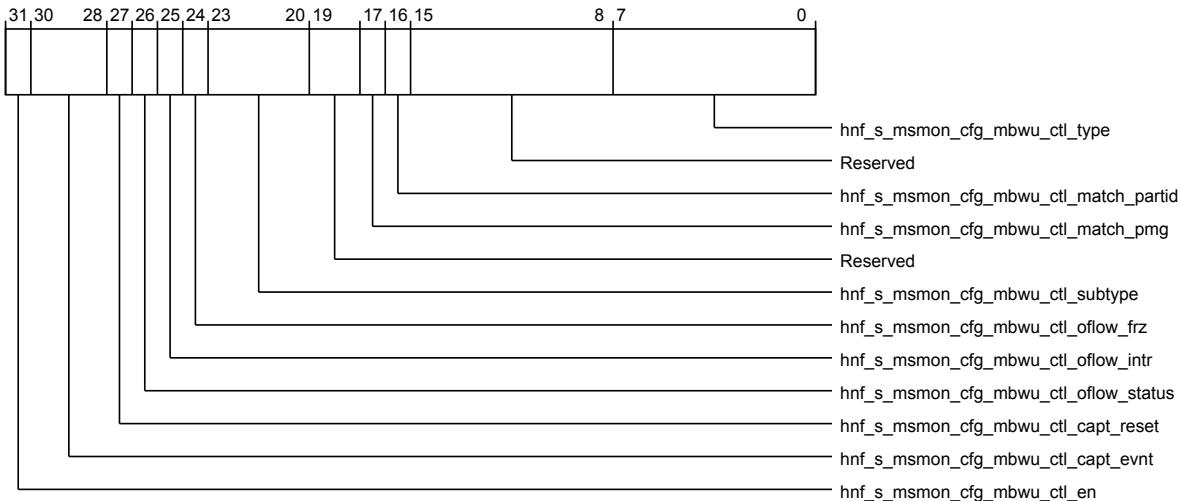


Figure 3-706 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_ctl (low)

The following table shows the por_hnf_s_msmon_cfg_mbwu_ctl lower register bit assignments.

Table 3-726 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_ctl (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_msmon_cfg_mbwu_ctl_en	0: The monitor is disabled and must not collect any information. 1: The monitor is enabled to collect information according to its configuration.	RW	1'h0
30:28	hnf_s_msmon_cfg_mbwu_ctl_capt_evnt	Select the event that triggers capture from the following: 0: No capture event is triggered. 1: External capture event 1 (optional but recommended)	RW	3'h0
27	hnf_s_msmon_cfg_mbwu_ctl_capt_reset	0: Monitor is not reset on capture. 1: Monitor is reset on capture.	RW	1'h0
26	hnf_s_msmon_cfg_mbwu_ctl_oflow_status	0: No overflow has occurred. 1: At least one overflow has occurred since this bit was last written.	RW	1'h0
25	hnf_s_msmon_cfg_mbwu_ctl_oflow_intr	0: No interrupt. 1: On overflow, an implementation-specific interrupt is signalled.	RW	1'h0

Table 3-726 por_hnf_mpam_s_por_hnf_s_msmon_cfg_mbwu_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
24	hnf_s_msmon_cfg_mbwu_ctl_oflow_frz	0: Monitor count wraps on overflow. 1: Monitor count freezes on overflow. The frozen value may be 0 or another value if the monitor overflowed with an increment larger than 1.	RW	1'h0
23:20	hnf_s_msmon_cfg_mbwu_ctl_subtype	A monitor can have other event matching criteria. The meaning of values in this field varies by monitor type. The MBWU monitor type supports: 0: Do not count any bandwidth. 1: Count bandwidth used by memory reads 2: Count bandwidth used by memory writes 3: Count bandwidth used by memory reads and memory writes All other values are reserved and behaviour of a monitor with SUBTYPE set to one of the reserved values is UNPREDICTABLE.	RW	4'h0
19:18	Reserved	Reserved	RO	-
17	hnf_s_msmon_cfg_mbwu_ctl_match_pmg	0: Monitor bandwidth used by all PMG values. 1: Only monitor bandwidth used with the PMG value matching MSMON_CFG_CSU_FLT.PMG.	RW	1'h0
16	hnf_s_msmon_cfg_mbwu_ctl_match_partid	0: Monitor bandwidth used by all PARTIDs. 1: Only monitor bandwidth used with the PARTID matching MSMON_CFG_MBWU_FLT.PARTID.	RW	1'h0
15:8	Reserved	Reserved	RO	-
7:0	hnf_s_msmon_cfg_mbwu_ctl_type	Read-only: Constant type indicating the type of the monitor. MBWU monitor is TYPE = 0x42.	RW	8'h42

por_hnf_s_msmon_csu

Memory system performance monitor cache storage usage monitor register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1840
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-707 por_hnf_mpam_s_por_hnf_s_msmon_csu (high)

The following table shows the por_hnf_s_msmon_csu higher register bit assignments.

Table 3-727 por_hnf_mpam_s_por_hnf_s_msmon_csu (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

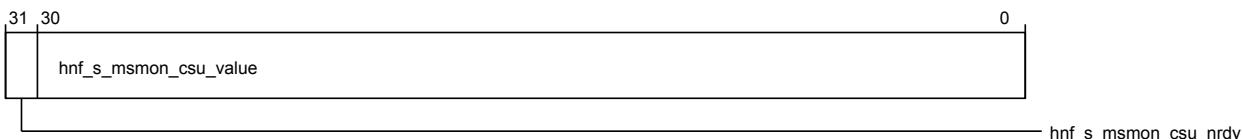


Figure 3-708 por_hnf_mpam_s_por_hnf_s_msmon_csu (low)

The following table shows the por_hnf_s_msmon_csu lower register bit assignments.

Table 3-728 por_hnf_mpam_s_por_hnf_s_msmon_csu (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_msmon_csu_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_s_msmon_csu_value	Cache storage usage value if NRDY is 0. Invalid if NRDY is 1. VALUE is the cache storage usage in bytes.	RW	31'h0

por_hnf_s_msmon_csu_capture

Memory system performance monitor cache storage usage capture register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1848

Register reset 64'b0

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

Secure group override por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

63	Reserved	32
----	----------	----

Figure 3-709 por_hnf_mpam_s_por_hnf_s_msmon_csu_capture (high)

The following table shows the por_hnf_s_msmon_csu_capture higher register bit assignments.

Table 3-729 por_hnf_mpam_s_por_hnf_s_msmon_csu_capture (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31 30	0
hnf_s_msmon_csu_vapture_value	
hnf_s_msmon_csu_capture_nrdy	

Figure 3-710 por_hnf_mpam_s_por_hnf_s_msmon_csu_capture (low)

The following table shows the por_hnf_s_msmon_csu_capture lower register bit assignments.

Table 3-730 por_hnf_mpam_s_por_hnf_s_msmon_csu_capture (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_msmon_csu_capture_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_s_msmon_csu_vapture_value	Cache storage usage value if NRDY is 0. Invalid if NRDY is 1. VALUE is the cache storage usage in bytes.	RW	31'h0

por_hnf_s_msmon_mbwu

Memory system performance monitor memory bandwidth usage monitor register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1860
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-711 por_hnf_mpam_s_por_hnf_s_msmon_mbwu (high)

The following table shows the por_hnf_s_msmon_mbwu higher register bit assignments.

Table 3-731 por_hnf_mpam_s_por_hnf_s_msmon_mbwu (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

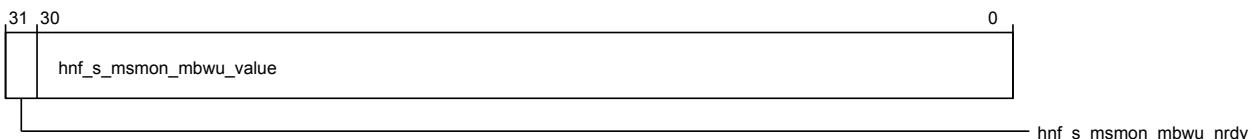


Figure 3-712 por_hnf_mpam_s_por_hnf_s_msmon_mbwu (low)

The following table shows the por_hnf_s_msmon_mbwu lower register bit assignments.

Table 3-732 por_hnf_mpam_s_por_hnf_s_msmon_mbwu (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_msmon_mbwu_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_s_msmon_mbwu_value	Memory channel bandwidth value if NRDY is 0. Invalid if NRDY is 1. VALUE is the memory channel bandwidth usage in megabytes.	RW	31'h0

por_hnf_s_msmon_mbwu_capture

Memory system performance monitor memory bandwidth usage capture register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1868

Register reset 64'b0

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

Secure group override por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-713 por_hnf_mpam_s_por_hnf_s_msmon_mbwu_capture (high)

The following table shows the por_hnf_s_msmon_mbwu_capture higher register bit assignments.

Table 3-733 por_hnf_mpam_s_por_hnf_s_msmon_mbwu_capture (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

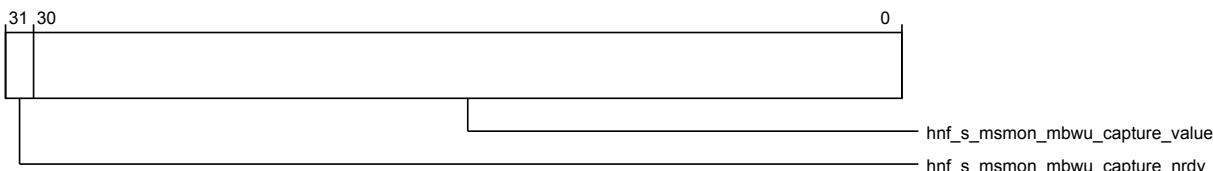


Figure 3-714 por_hnf_mpam_s_por_hnf_s_msmon_mbwu_capture (low)

The following table shows the por_hnf_s_msmon_mbwu_capture lower register bit assignments.

Table 3-734 por_hnf_mpam_s_por_hnf_s_msmon_mbwu_capture (low)

Bits	Field name	Description	Type	Reset
31	hnf_s_msmon_mbwu_capture_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_s_msmon_mbwu_capture_value	Memory channel bandwidth value if NRDY is 0. Invalid if NRDY is 1. VALUE is the memory channel bandwidth usage in megabytes.	RW	31'h0

por_hnf_s_mpamcfg_cpbm

MPAM cache portion bitmap partition configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2000
Register reset	64'b1111111111111111
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.
Secure group override	por_hnf_mpam_s_secure_register_groups_override.mpam

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-715 por_hnf_mpam_s_por_hnf_s_mpamcfg_cpbm (high)

The following table shows the por_hnf_s_mpamcfg_cpbm higher register bit assignments.

Table 3-735 por_hnf_mpam_s_por_hnf_s_mpamcfg_cpbm (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31

16 15

0

Reserved

hnf_s_mpamcfg_cpbm_cpbm

Figure 3-716 por_hnf_mpam_s_por_hnf_s_mpamcfg_cpbm (low)

The following table shows the por_hnf_s_mpamcfg_cpbm lower register bit assignments.

Table 3-736 por_hnf_mpam_s_por_hnf_s_mpamcfg_cpbm (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	hnf_s_mpamcfg_cpbm_cpbm	Bitmap of portions of cache capacity allocable by the partition selected by MPAMCFG_PART_SEL. NOTE: CPBM can not be all zeros for any PARTID.	RW	16'hFFFF

3.3.12 Configuration master register descriptions

This section lists the configuration registers.

por_cfgm_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

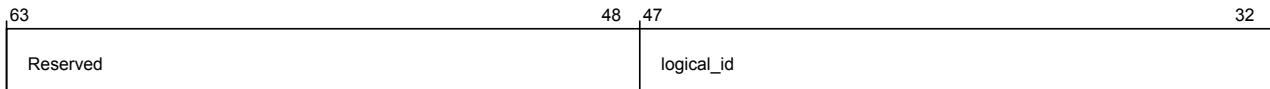


Figure 3-717 por_cfgm_por_cfgm_node_info (high)

The following table shows the por_cfgm_node_info higher register bit assignments.

Table 3-737 por_cfgm_por_cfgm_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

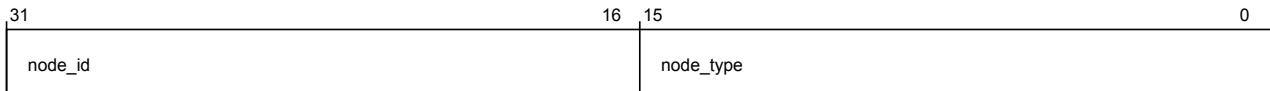


Figure 3-718 por_cfgm_por_cfgm_node_info (low)

The following table shows the por_cfgm_node_info lower register bit assignments.

Table 3-738 por_cfgm_por_cfgm_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0002

por_cfgm_periph_id_0_periph_id_1

Functions as the peripheral ID 0 and peripheral ID 1 register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h8

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

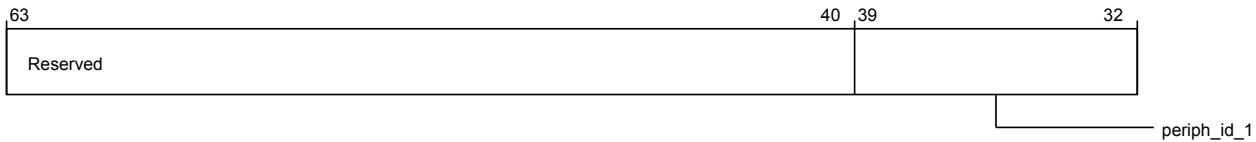


Figure 3-719 por_cfgm_por_cfgm_periph_id_0_periph_id_1 (high)

The following table shows the por_cfgm_periph_id_0_periph_id_1 higher register bit assignments.

Table 3-739 por_cfgm_por_cfgm_periph_id_0_periph_id_1 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	periph_id_1	Peripheral ID 1	RO	8'b10110100

The following image shows the lower register bit assignments.

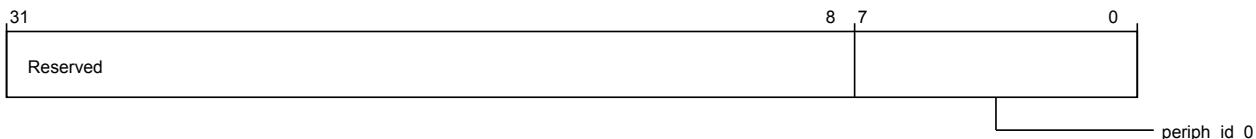


Figure 3-720 por_cfgm_por_cfgm_periph_id_0_periph_id_1 (low)

The following table shows the por_cfgm_periph_id_0_periph_id_1 lower register bit assignments.

Table 3-740 por_cfgm_por_cfgm_periph_id_0_periph_id_1 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	periph_id_0	Peripheral ID 0	RO	Configuration dependent

por_cfgm_periph_id_2_periph_id_3

Functions as the peripheral ID 2 and peripheral ID 3 register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h10

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

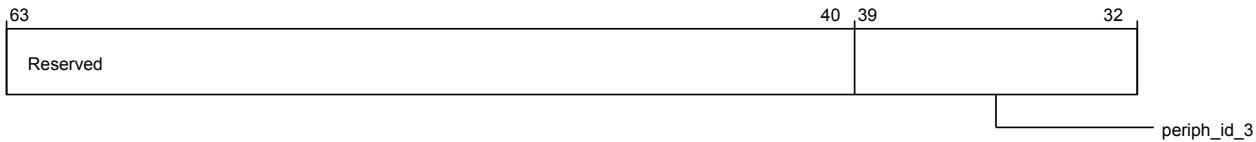


Figure 3-721 por_cfgm_por_cfgm_periph_id_2_periph_id_3 (high)

The following table shows the por_cfgm_periph_id_2_periph_id_3 higher register bit assignments.

Table 3-741 por_cfgm_por_cfgm_periph_id_2_periph_id_3 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	periph_id_3	Peripheral ID 3	RO	8'b0

The following image shows the lower register bit assignments.

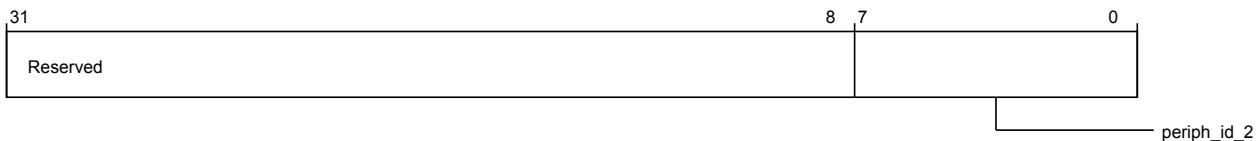


Figure 3-722 por_cfgm_por_cfgm_periph_id_2_periph_id_3 (low)

The following table shows the por_cfgm_periph_id_2_periph_id_3 lower register bit assignments.

Table 3-742 por_cfgm_por_cfgm_periph_id_2_periph_id_3 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	periph_id_2	Peripheral ID 2 [7:4] indicates revision: 0x0 r0p0 0x1 r1p0 0x2 r1p1 0x3 r2p0 [3] JEDEC JEP106 identity code, 1'b1 [2:0] JEP106 identity code [6:4], 0'b011	RO	Configuration dependent

por_cfgm_periph_id_4_periph_id_5

Functions as the peripheral ID 4 and peripheral ID 5 register.

Its characteristics are:

Type RO
Register width (Bits) 64

Address offset	16'h18
Register reset	64'b011000100
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

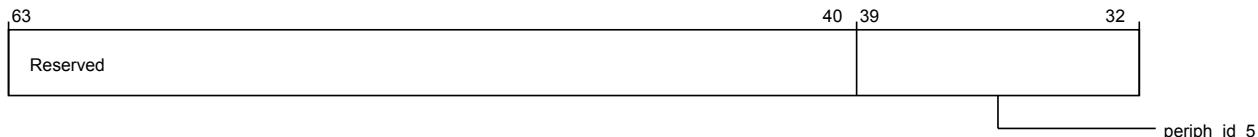


Figure 3-723 por_cfgm_por_cfgm_periph_id_4_periph_id_5 (high)

The following table shows the por_cfgm_periph_id_4_periph_id_5 higher register bit assignments.

Table 3-743 por_cfgm_por_cfgm_periph_id_4_periph_id_5 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	periph_id_5	Peripheral ID 5	RO	8'b0

The following image shows the lower register bit assignments.

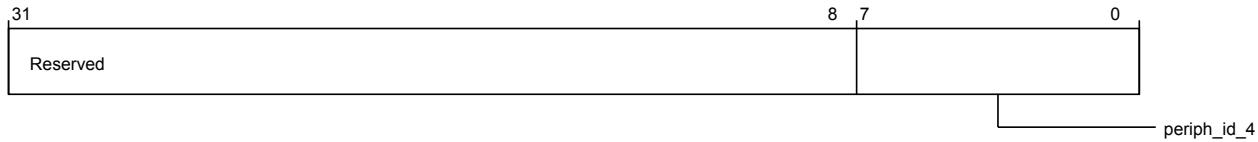


Figure 3-724 por_cfgm_por_cfgm_periph_id_4_periph_id_5 (low)

The following table shows the por_cfgm_periph_id_4_periph_id_5 lower register bit assignments.

Table 3-744 por_cfgm_por_cfgm_periph_id_4_periph_id_5 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	periph_id_4	Peripheral ID 4	RO	8'b11000100

por_cfgm_periph_id_6_periph_id_7

Functions as the peripheral ID 6 and peripheral ID 7 register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h20
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

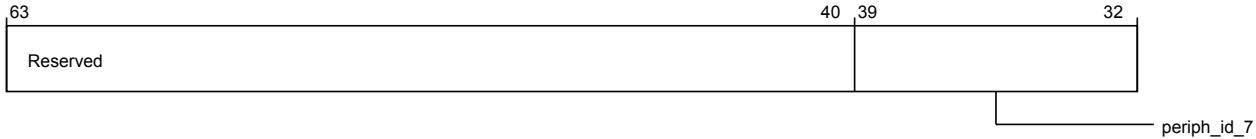


Figure 3-725 por_cfgm_por_cfgm_periph_id_6_periph_id_7 (high)

The following table shows the por_cfgm_periph_id_6_periph_id_7 higher register bit assignments.

Table 3-745 por_cfgm_por_cfgm_periph_id_6_periph_id_7 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	periph_id_7	Peripheral ID 7	RO	8'b0

The following image shows the lower register bit assignments.

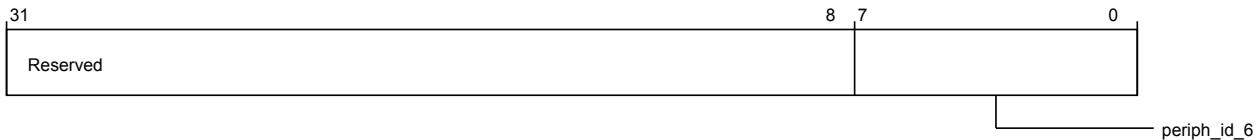


Figure 3-726 por_cfgm_por_cfgm_periph_id_6_periph_id_7 (low)

The following table shows the por_cfgm_periph_id_6_periph_id_7 lower register bit assignments.

Table 3-746 por_cfgm_por_cfgm_periph_id_6_periph_id_7 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	periph_id_6	Peripheral ID 6	RO	8'b0

por_cfgm_component_id_0_component_id_1

Functions as the component ID 0 and component ID 1 register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h28

Register reset 64'b1111000000001101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

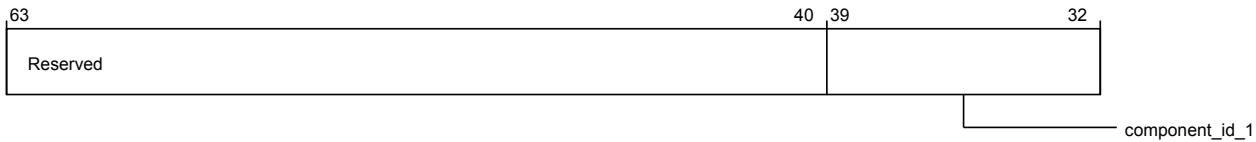


Figure 3-727 por_cfgm_por_cfgm_component_id_0_component_id_1 (high)

The following table shows the por_cfgm_component_id_0_component_id_1 higher register bit assignments.

Table 3-747 por_cfgm_por_cfgm_component_id_0_component_id_1 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	component_id_1	Component ID 1	RO	8'b11110000

The following image shows the lower register bit assignments.

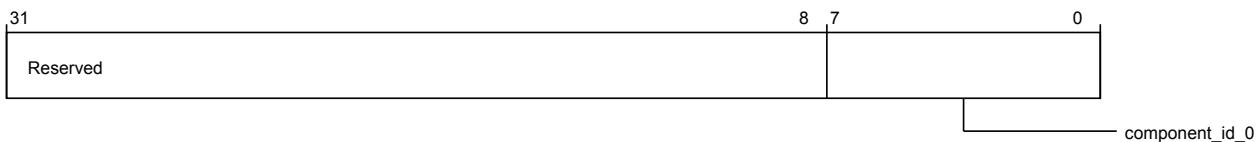


Figure 3-728 por_cfgm_por_cfgm_component_id_0_component_id_1 (low)

The following table shows the por_cfgm_component_id_0_component_id_1 lower register bit assignments.

Table 3-748 por_cfgm_por_cfgm_component_id_0_component_id_1 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	component_id_0	Component ID 0	RO	8'b00001101

por_cfgm_component_id_2_component_id_3

Functions as the component ID 2 and component ID 3 register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h30

Register reset 64'b1011000100000101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

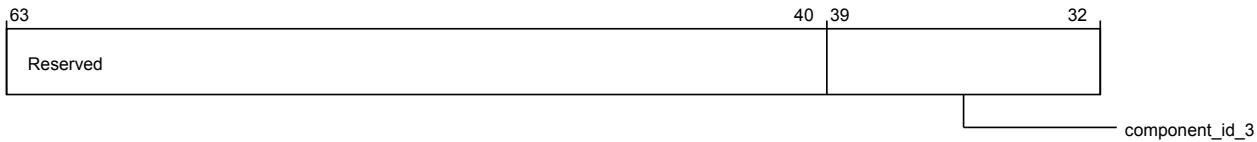


Figure 3-729 por_cfgm_por_cfgm_component_id_2_component_id_3 (high)

The following table shows the por_cfgm_component_id_2_component_id_3 higher register bit assignments.

Table 3-749 por_cfgm_por_cfgm_component_id_2_component_id_3 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	component_id_3	Component ID 3	RO	8'b10110001

The following image shows the lower register bit assignments.

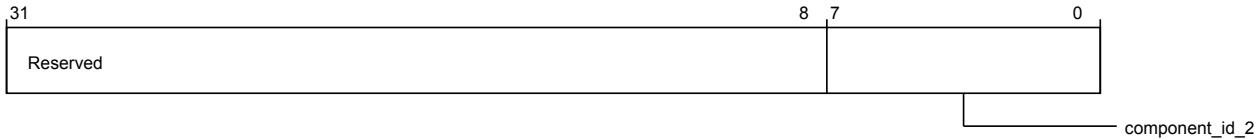


Figure 3-730 por_cfgm_por_cfgm_component_id_2_component_id_3 (low)

The following table shows the por_cfgm_component_id_2_component_id_3 lower register bit assignments.

Table 3-750 por_cfgm_por_cfgm_component_id_2_component_id_3 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	component_id_2	Component ID 2	RO	8'b00000101

por_cfgm_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

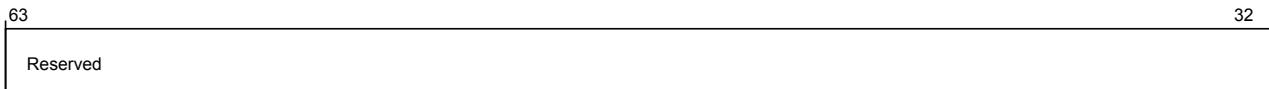


Figure 3-731 por_cfgm_por_cfgm_child_info (high)

The following table shows the por_cfgm_child_info higher register bit assignments.

Table 3-751 por_cfgm_por_cfgm_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

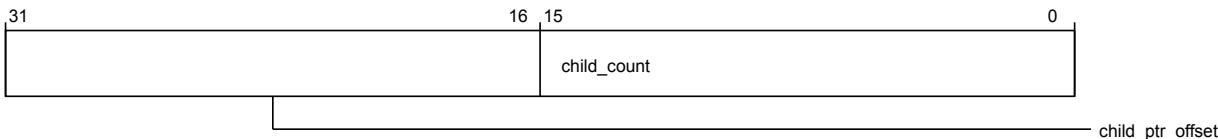


Figure 3-732 por_cfgm_por_cfgm_child_info (low)

The following table shows the por_cfgm_child_info lower register bit assignments.

Table 3-752 por_cfgm_por_cfgm_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h100
15:0	child_count	Number of child nodes; used in discovery process	RO	Configuration dependent

por_cfgm_secure_access

Functions as the secure access control register. This register must be set up at boot time. Before initiating a write to this register, software must ensure that no other configuration accesses are in flight. Once this write is initiated, no other configuration accesses are initiated until complete.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-733 por_cfgm_por_cfgm_secure_access (high)

The following table shows the por_cfgm_secure_access higher register bit assignments.

Table 3-753 por_cfgm_por_cfgm_secure_access (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

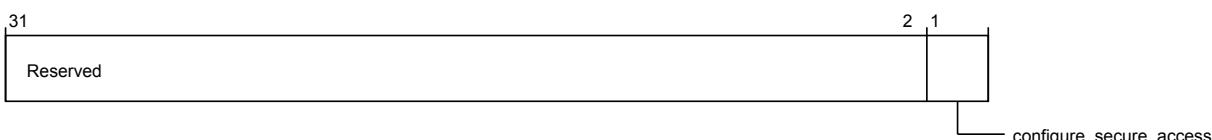


Figure 3-734 por_cfgm_por_cfgm_secure_access (low)

The following table shows the por_cfgm_secure_access lower register bit assignments.

Table 3-754 por_cfgm_por_cfgm_secure_access (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1:0	configure_secure_access	Secure access mode 2'b00: Default operation 2'b01: Allows non-secure access to secure registers 2'b10: Allows secure access only to any configuration register regardless of its security status 2'b11: Undefined behavior	RW	2'b0

por_cfgm_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h988
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-735 por_cfgm_por_cfgm_secure_register_groups_override (high)

The following table shows the por_cfgm_secure_register_groups_override higher register bit assignments.

Table 3-755 por_cfgm_por_cfgm_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

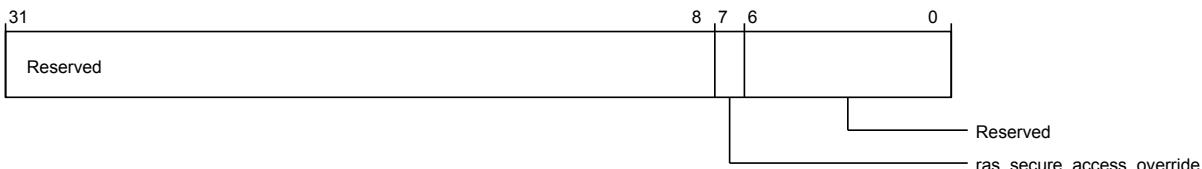


Figure 3-736 por_cfgm_por_cfgm_secure_register_groups_override (low)

The following table shows the por_cfgm_secure_register_groups_override lower register bit assignments.

Table 3-756 por_cfgm_por_cfgm_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0
6:0	Reserved	Reserved	RO	-

por_cfgm_errgsr_mxp_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the XP <n> secure #{{map[index%2]}} status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3000 + #{{8*[0, 1, 2, 3, 4, 5, 6, 7]}}

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_cfgm_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

63

32

err_status_{index}

Figure 3-737 por_cfgm_por_cfgm_errgsr_mxp_0-7 (high)

The following table shows the por_cfgm_errgsr_mxp_0-7 higher register bit assignments.

Table 3-757 por_cfgm_por_cfgm_errgsr_mxp_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}	Read-only copy of MXP #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

31

0

err_status_{index}

Figure 3-738 por_cfgm_por_cfgm_errgsr_mxp_0-7 (low)

The following table shows the por_cfgm_errgsr_mxp_0-7 lower register bit assignments.

Table 3-758 por_cfgm_por_cfgm_errgsr_mxp_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}	Read-only copy of MXP #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_mxp_0-7_NS

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the XP <n> non-secure #{map[index%2]} status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3040 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

err_status_{index}_ns

Figure 3-739 por_cfgm_por_cfgm_errgsr_mxp_0-7_ns (high)

The following table shows the por_cfgm_errgsr_mxp_0-7_NS higher register bit assignments.

Table 3-759 por_cfgm_por_cfgm_errgsr_mxp_0-7_ns (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}_ns	Read-only copy of MXP #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

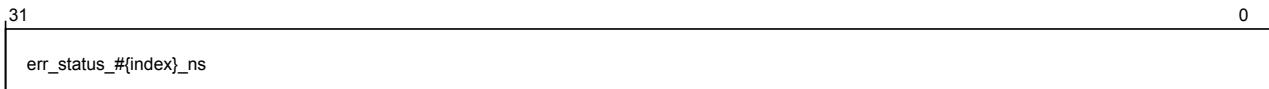


Figure 3-740 por_cfgm_por_cfgm_errgsr_mxp_0-7_ns (low)

The following table shows the por_cfgm_errgsr_mxp_0-7_NS lower register bit assignments.

Table 3-760 por_cfgm_por_cfgm_errgsr_mxp_0-7_ns (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}_ns	Read-only copy of MXP #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_hni_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the HNI <n> secure #{map[index%2]} status.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3080 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cfgm_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

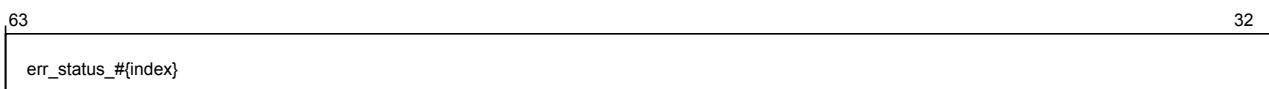


Figure 3-741 por_cfgm_por_cfgm_errgsr_hni_0-7 (high)

The following table shows the por_cfgm_errgsr_hni_0-7 higher register bit assignments.

Table 3-761 por_cfgm_por_cfgm_errgsr_hni_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}	Read-only copy of HNI #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.



Figure 3-742 por_cfgm_por_cfgm_errgsr_hni_0-7 (low)

The following table shows the por_cfgm_errgsr_hni_0-7 lower register bit assignments.

Table 3-762 por_cfgm_por_cfgm_errgsr_hni_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}	Read-only copy of HNI # {map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_hni_0-7_NS

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the HNI <n> non-secure # {map[index%2]} status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h30C0 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

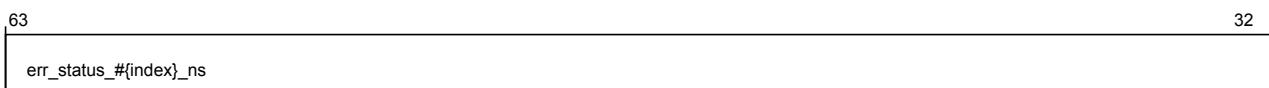


Figure 3-743 por_cfgm_por_cfgm_errgsr_hni_0-7_ns (high)

The following table shows the por_cfgm_errgsr_hni_0-7_NS higher register bit assignments.

Table 3-763 por_cfgm_por_cfgm_errgsr_hni_0-7_ns (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}_ns	Read-only copy of HNI # {map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

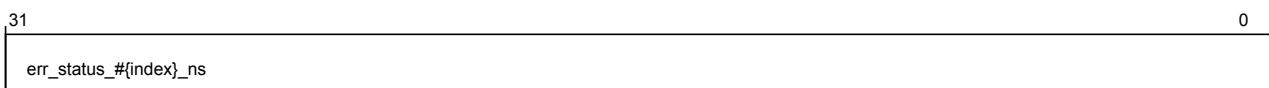


Figure 3-744 por_cfgm_por_cfgm_errgsr_hni_0-7_ns (low)

The following table shows the por_cfgm_errgsr_hni_0-7_NS lower register bit assignments.

Table 3-764 por_cfgm_por_cfgm_errgsr_hni_0-7_ns (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}_ns	Read-only copy of HNI #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_hnf_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the HNF <n> secure #{map[index%2]} status.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3100 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cfgm_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

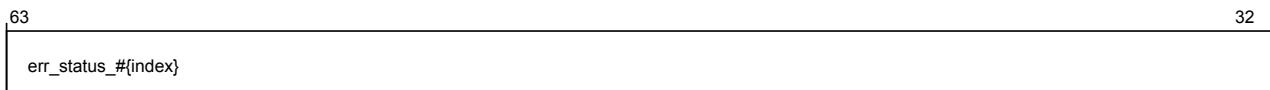


Figure 3-745 por_cfgm_por_cfgm_errgsr_hnf_0-7 (high)

The following table shows the por_cfgm_errgsr_hnf_0-7 higher register bit assignments.

Table 3-765 por_cfgm_por_cfgm_errgsr_hnf_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}	Read-only copy of HNF #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

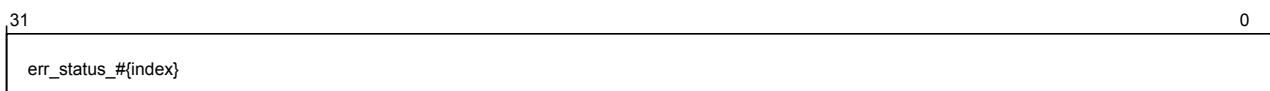


Figure 3-746 por_cfgm_por_cfgm_errgsr_hnf_0-7 (low)

The following table shows the por_cfgm_errgsr_hnf_0-7 lower register bit assignments.

Table 3-766 por_cfgm_por_cfgm_errgsr_hnf_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}	Read-only copy of HNF #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_hnf_0-7_NS

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the HNF <n> non-secure # $\{\text{map}[\text{index}\%2]\}$ status.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3140 + # $\{8*[0, 1, 2, 3, 4, 5, 6, 7]\}$
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-747 por_cfgm_por_cfgm_errgsr_hnf_0-7_ns (high)

The following table shows the por_cfgm_errgsr_hnf_0-7_NS higher register bit assignments.

Table 3-767 por_cfgm_por_cfgm_errgsr_hnf_0-7_ns (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}_ns	Read-only copy of HNF # $\{\text{map}[\text{index}\%2]\}$ <n> status	RO	64'h0

The following image shows the lower register bit assignments.

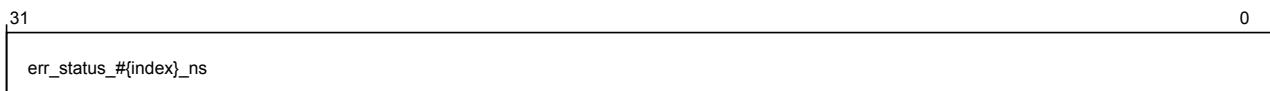


Figure 3-748 por_cfgm_por_cfgm_errgsr_hnf_0-7_ns (low)

The following table shows the por_cfgm_errgsr_hnf_0-7_NS lower register bit assignments.

Table 3-768 por_cfgm_por_cfgm_errgsr_hnf_0-7_ns (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}_ns	Read-only copy of HNF # $\{\text{map}[\text{index}\%2]\}$ <n> status	RO	64'h0

por_cfgm_errgsr_sbsx_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the SBSX <n> secure # $\{\text{map}[\text{index}\%2]\}$ status.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3180 + # $\{8*[0, 1, 2, 3, 4, 5, 6, 7]\}$

Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cfgm_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-749 por_cfgm_por_cfgm_errgsr_sbsx_0-7 (high)

The following table shows the por_cfgm_errgsr_sbsx_0-7 higher register bit assignments.

Table 3-769 por_cfgm_por_cfgm_errgsr_sbsx_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}	Read-only copy of SBSX #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

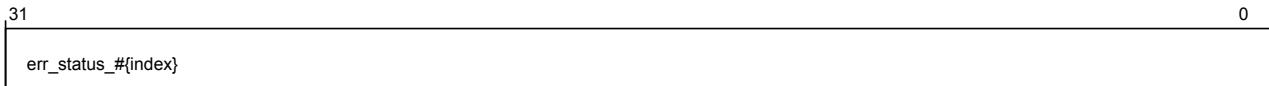


Figure 3-750 por_cfgm_por_cfgm_errgsr_sbsx_0-7 (low)

The following table shows the por_cfgm_errgsr_sbsx_0-7 lower register bit assignments.

Table 3-770 por_cfgm_por_cfgm_errgsr_sbsx_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}	Read-only copy of SBSX #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_sbsx_0-7_NS

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the SBSX <n> non-secure #{map[index%2]} status.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h31C0 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

err_status_{index}_ns

Figure 3-751 por_cfgm_por_cfgm_errgsr_sbsx_0-7_ns (high)

The following table shows the por_cfgm_errgsr_sbsx_0-7_NS higher register bit assignments.

Table 3-771 por_cfgm_por_cfgm_errgsr_sbsx_0-7_ns (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}_ns	Read-only copy of SBSX #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

31

0

err_status_{index}_ns

Figure 3-752 por_cfgm_por_cfgm_errgsr_sbsx_0-7_ns (low)

The following table shows the por_cfgm_errgsr_sbsx_0-7_NS lower register bit assignments.

Table 3-772 por_cfgm_por_cfgm_errgsr_sbsx_0-7_ns (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}_ns	Read-only copy of SBSX #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_cxg_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the CXG <n> secure #{map[index%2]} status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3200 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group por_cfgm_secure_register_groups_override.ras_secure_access_override
override

The following image shows the higher register bit assignments.

63

32

err_status_{index}

Figure 3-753 por_cfgm_por_cfgm_errgsr_cxg_0-7 (high)

The following table shows the por_cfgm_errgsr_cxg_0-7 higher register bit assignments.

Table 3-773 por_cfgm_por_cfgm_errgsr_cxg_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}	Read-only copy of CXG #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

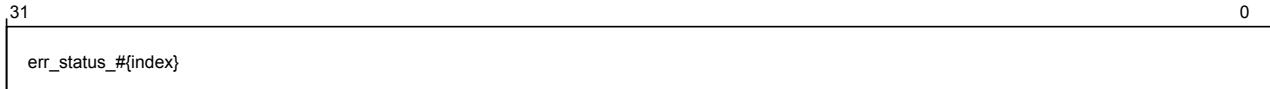


Figure 3-754 por_cfgm_por_cfgm_errgsr_cxg_0-7 (low)

The following table shows the por_cfgm_errgsr_cxg_0-7 lower register bit assignments.

Table 3-774 por_cfgm_por_cfgm_errgsr_cxg_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}	Read-only copy of CXG #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_cxg_0-7_NS

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the CXG <n> non-secure #'{map[index%2]} status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3240 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

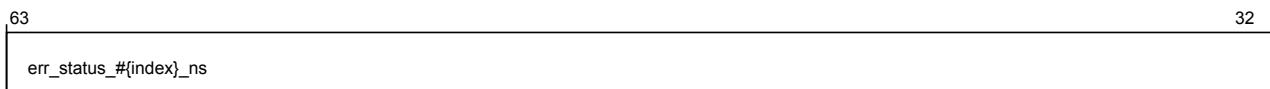


Figure 3-755 por_cfgm_por_cfgm_errgsr_cxg_0-7_ns (high)

The following table shows the por_cfgm_errgsr_cxg_0-7_NS higher register bit assignments.

Table 3-775 por_cfgm_por_cfgm_errgsr_cxg_0-7_ns (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}_ns	Read-only copy of CXG #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

31	err_status_{index}_ns	0
----	-----------------------	---

Figure 3-756 por_cfgm_por_cfgm_errgsr_cxg_0-7_ns (low)

The following table shows the por_cfgm_errgsr_cxg_0-7_NS lower register bit assignments.

Table 3-776 por_cfgm_por_cfgm_errgsr_cxg_0-7_ns (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}_ns	Read-only copy of CXG # {map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_mtsx_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the MTSX <n> secure # {map[index%2]} status.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3280 + # {8*[0, 1, 2, 3, 4, 5, 6, 7]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_cfgm_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

63	err_status_{index}	32
----	--------------------	----

Figure 3-757 por_cfgm_por_cfgm_errgsr_mtsx_0-7 (high)

The following table shows the por_cfgm_errgsr_mtsx_0-7 higher register bit assignments.

Table 3-777 por_cfgm_por_cfgm_errgsr_mtsx_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}	Read-only copy of MTSX # {map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

31	err_status_{index}	0
----	--------------------	---

Figure 3-758 por_cfgm_por_cfgm_errgsr_mtsx_0-7 (low)

The following table shows the por_cfgm_errgsr_mtsx_0-7 lower register bit assignments.

Table 3-778 por_cfgm_por_cfgm_errgsr_mtsx_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}	Read-only copy of MTSX #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errgsr_mtsx_0-7_NS

This register repeats 7 times. It parametrized by the index from 0 to 7. Provides the MTSX <n> non-secure #{map[index%2]} status.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h32C0 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

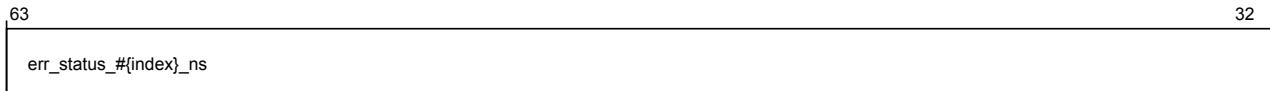


Figure 3-759 por_cfgm_por_cfgm_errgsr_mtsx_0-7_ns (high)

The following table shows the por_cfgm_errgsr_mtsx_0-7_NS higher register bit assignments.

Table 3-779 por_cfgm_por_cfgm_errgsr_mtsx_0-7_ns (high)

Bits	Field name	Description	Type	Reset
63:32	err_status_{index}_ns	Read-only copy of MTSX #{map[index%2]} <n> status	RO	64'h0

The following image shows the lower register bit assignments.

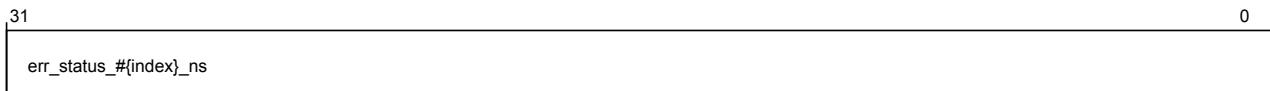


Figure 3-760 por_cfgm_por_cfgm_errgsr_mtsx_0-7_ns (low)

The following table shows the por_cfgm_errgsr_mtsx_0-7_NS lower register bit assignments.

Table 3-780 por_cfgm_por_cfgm_errgsr_mtsx_0-7_ns (low)

Bits	Field name	Description	Type	Reset
31:0	err_status_{index}_ns	Read-only copy of MTSX #{map[index%2]} <n> status	RO	64'h0

por_cfgm_errdevaff

Functions as the device affinity register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3FA8
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

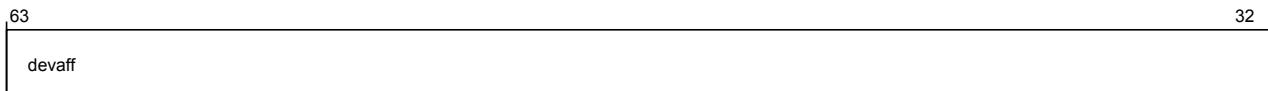


Figure 3-761 por_cfgm_por_cfgm_errdevaff (high)

The following table shows the por_cfgm_errdevaff higher register bit assignments.

Table 3-781 por_cfgm_por_cfgm_errdevaff (high)

Bits	Field name	Description	Type	Reset
63:32	devaff	Device affinity register	RO	64'b0

The following image shows the lower register bit assignments.

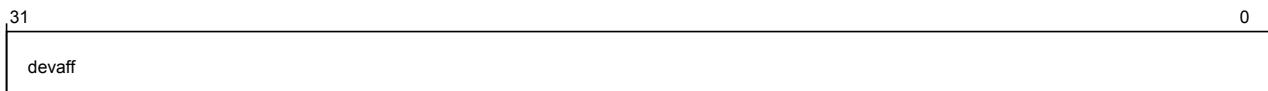


Figure 3-762 por_cfgm_por_cfgm_errdevaff (low)

The following table shows the por_cfgm_errdevaff lower register bit assignments.

Table 3-782 por_cfgm_por_cfgm_errdevaff (low)

Bits	Field name	Description	Type	Reset
31:0	devaff	Device affinity register	RO	64'b0

por_cfgm_errdevarch

Functions as the device architecture register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3FB8
Register reset	64'b0001011101110000000000101000

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

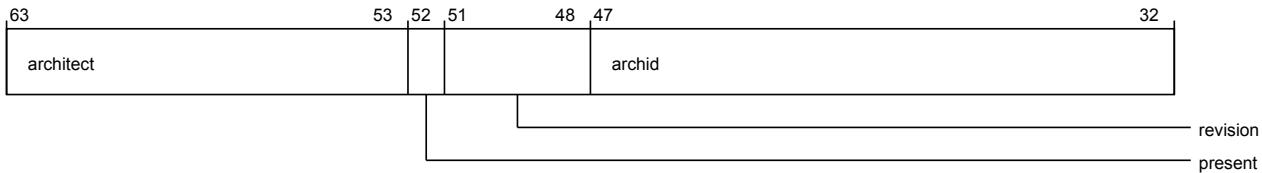


Figure 3-763 por_cfgm_por_cfgm_errdevarch (high)

The following table shows the por_cfgm_errdevarch higher register bit assignments.

Table 3-783 por_cfgm_por_cfgm_errdevarch (high)

Bits	Field name	Description	Type	Reset
63:53	architect	Architect	RO	11'h23B
52	present	Present	RO	1'b1
51:48	revision	Architecture revision	RO	4'b0
47:32	archid	Architecture ID	RO	16'h0A00

The following image shows the lower register bit assignments.

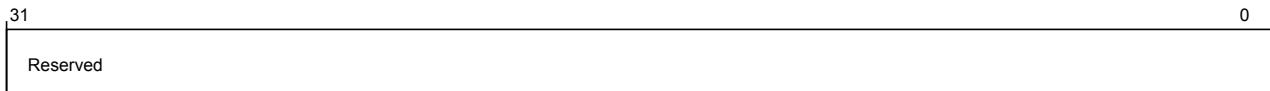


Figure 3-764 por_cfgm_por_cfgm_errdevarch (low)

The following table shows the por_cfgm_errdevarch lower register bit assignments.

Table 3-784 por_cfgm_por_cfgm_errdevarch (low)

Bits	Field name	Description	Type	Reset
31:0	Reserved	Reserved	RO	-

por_cfgm_erridr

Contains the number of error records.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3FC8

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

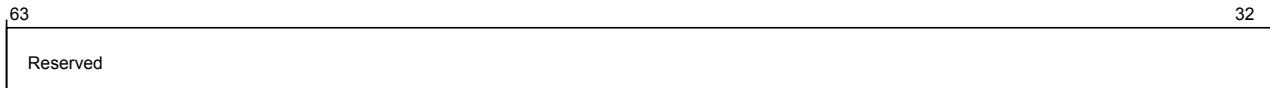


Figure 3-765 por_cfgm_por_cfgm_erridr (high)

The following table shows the por_cfgm_erridr higher register bit assignments.

Table 3-785 por_cfgm_por_cfgm_erridr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

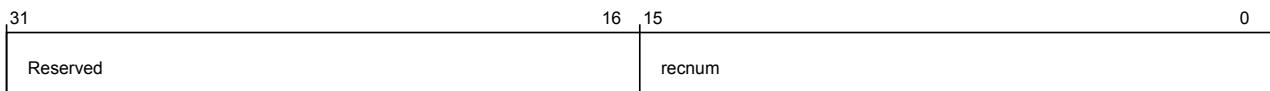


Figure 3-766 por_cfgm_por_cfgm_erridr (low)

The following table shows the por_cfgm_erridr lower register bit assignments.

Table 3-786 por_cfgm_por_cfgm_erridr (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	recnum	Number of error records; equal to 2*(number of logical devices)	RO	Configuration dependent

por_cfgm_errpidr45

Functions as the identification register for peripheral ID 4 and peripheral ID 5.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3FD0

Register reset 64'b000000100

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

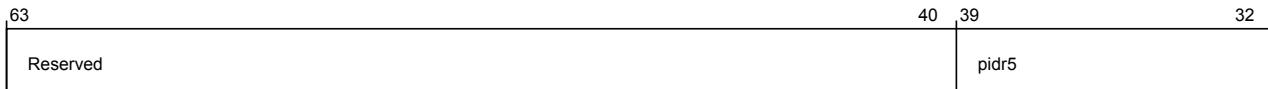


Figure 3-767 por_cfgm_por_cfgm_errpidr45 (high)

The following table shows the por_cfgm_por_cfgm_errpidr45 higher register bit assignments.

Table 3-787 por_cfgm_por_cfgm_errpidr45 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr5	Peripheral ID 5	RO	8'b0

The following image shows the lower register bit assignments.

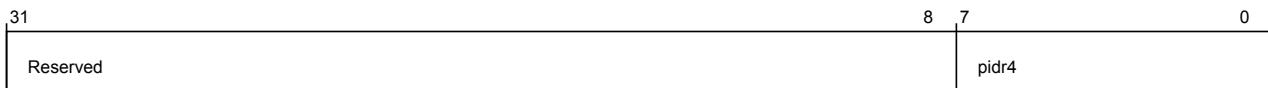


Figure 3-768 por_cfgm_por_cfgm_errpidr45 (low)

The following table shows the por_cfgm_por_cfgm_errpidr45 lower register bit assignments.

Table 3-788 por_cfgm_por_cfgm_errpidr45 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr4	Peripheral ID 4	RO	8'h4

por_cfgm_errpidr67

Functions as the identification register for peripheral ID 6 and peripheral ID 7.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3FD8

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

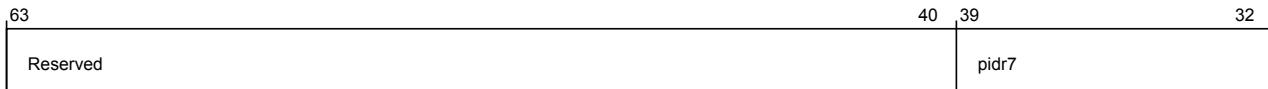


Figure 3-769 por_cfgm_por_cfgm_errpidr67 (high)

The following table shows the por_cfgm_por_cfgm_errpidr67 higher register bit assignments.

Table 3-789 por_cfgm_por_cfgm_errpidr67 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr7	Peripheral ID 7	RO	8'b0

The following image shows the lower register bit assignments.

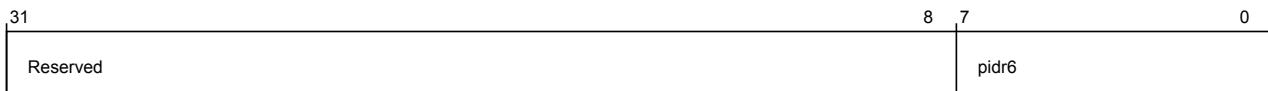


Figure 3-770 por_cfgm_por_cfgm_errpidr67 (low)

The following table shows the por_cfgm_por_cfgm_errpidr67 lower register bit assignments.

Table 3-790 por_cfgm_por_cfgm_errpidr67 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr6	Peripheral ID 6	RO	8'b0

por_cfgm_errpidr01

Functions as the identification register for peripheral ID 0 and peripheral ID 1.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3FE0

Register reset 64'b0101110000011100

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

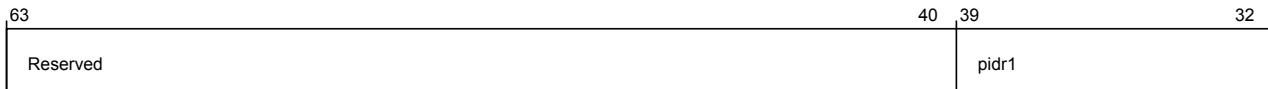


Figure 3-771 por_cfgm_por_cfgm_errpidr01 (high)

The following table shows the por_cfgm_por_cfgm_errpidr01 higher register bit assignments.

Table 3-791 por_cfgm_por_cfgm_errpidr01 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr1	Peripheral ID 1	RO	8'hb4

The following image shows the lower register bit assignments.

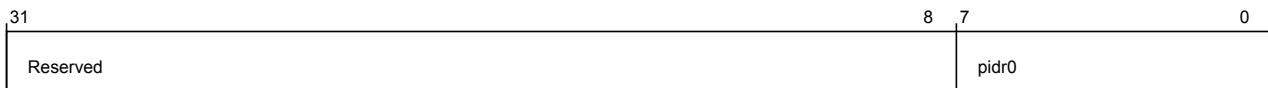


Figure 3-772 por_cfgm_por_cfgm_errpidr01 (low)

The following table shows the por_cfgm_por_cfgm_errpidr01 lower register bit assignments.

Table 3-792 por_cfgm_por_cfgm_errpidr01 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr0	Peripheral ID 0	RO	8'h34

por_cfgm_errpidr23

Functions as the identification register for peripheral ID 2 and peripheral ID 3.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3FE8

Register reset 64'b0000000111

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

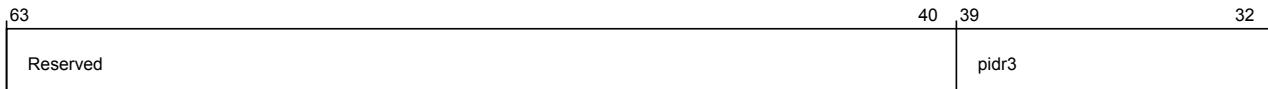


Figure 3-773 por_cfgm_por_cfgm_errpidr23 (high)

The following table shows the por_cfgm_por_cfgm_errpidr23 higher register bit assignments.

Table 3-793 por_cfgm_por_cfgm_errpidr23 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr3	Peripheral ID 3	RO	8'b0

The following image shows the lower register bit assignments.

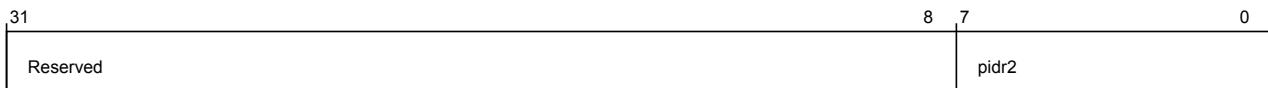


Figure 3-774 por_cfgm_por_cfgm_errpidr23 (low)

The following table shows the por_cfgm_por_cfgm_errpidr23 lower register bit assignments.

Table 3-794 por_cfgm_por_cfgm_errpidr23 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr2	Peripheral ID 2	RO	8'h7

por_cfgm_errcidr01

Functions as the identification register for component ID 0 and component ID 1.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3FF0

Register reset 64'b1111111100001101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

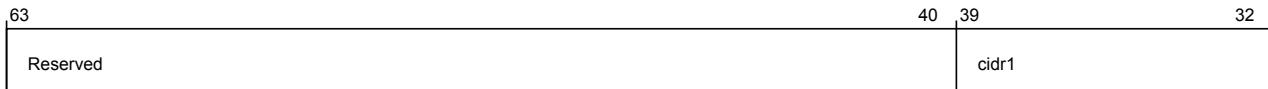


Figure 3-775 por_cfgm_por_cfgm_errcidr01 (high)

The following table shows the por_cfgm_errcidr01 higher register bit assignments.

Table 3-795 por_cfgm_por_cfgm_errcidr01 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	cidr1	Component ID 1	RO	8'hff

The following image shows the lower register bit assignments.

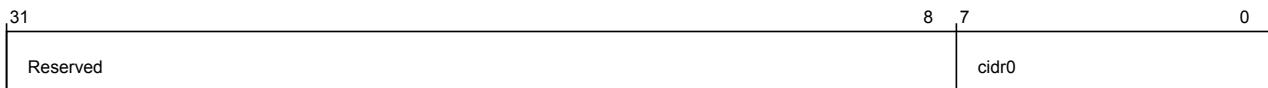


Figure 3-776 por_cfgm_por_cfgm_errcidr01 (low)

The following table shows the por_cfgm_errcidr01 lower register bit assignments.

Table 3-796 por_cfgm_por_cfgm_errcidr01 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	cidr0	Component ID 0	RO	8'hd

por_cfgm_errcidr23

Functions as the identification register for component ID 2 and component ID 3.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3FF8

Register reset 64'b0001011100000101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

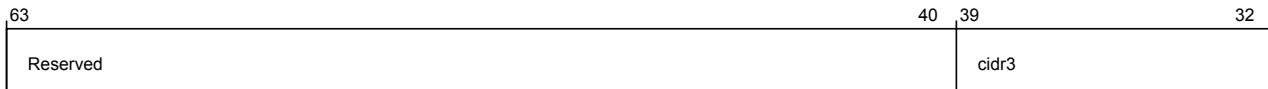


Figure 3-777 por_cfgm_por_cfgm_errcidr23 (high)

The following table shows the por_cfgm_errcidr23 higher register bit assignments.

Table 3-797 por_cfgm_por_cfgm_errcidr23 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	cidr3	Component ID 3	RO	8'hb1

The following image shows the lower register bit assignments.

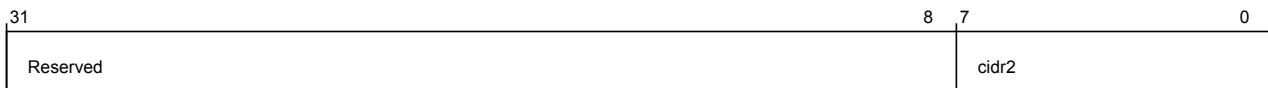


Figure 3-778 por_cfgm_por_cfgm_errcidr23 (low)

The following table shows the por_cfgm_errcidr23 lower register bit assignments.

Table 3-798 por_cfgm_por_cfgm_errcidr23 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	cidr2	Component ID 2	RO	8'h5

por_info_global

Contains user-specified values of build-time global configuration parameters.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

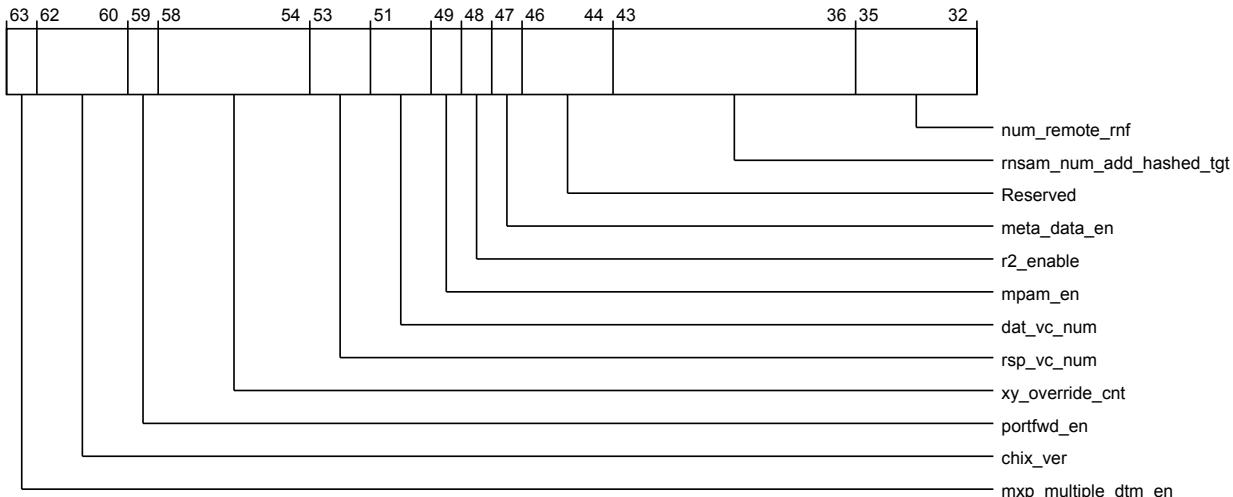


Figure 3-779 por_cfgm_por_info_global (high)

The following table shows the por_info_global higher register bit assignments.

Table 3-799 por_cfgm_por_info_global (high)

Bits	Field name	Description	Type	Reset
63	mfp_multiple_dtm_en	Multiple DTMs feature enable. This is used if number of device ports on the XP is > 2	RO	Configuration dependent
62:60	chix_ver	CHIX Version Parameter: 2 -> CHIB, 3 -> CHIC, 4 -> CHID, 5 -> CHIE	RO	Configuration dependent
59	portfwd_en	CCIX Port to Port Forwarding feature enable	RO	Configuration dependent
58:54	xy_override_cnt	Number of Src-Tgt pairs whose XY route path can be overridden	RO	Configuration dependent
53:52	rsp_vc_num	Number of additional RSP channels internal to MXP. For increased bandwidth, this parameter need to be set to 2	RO	Configuration dependent
51:50	dat_vc_num	Number of additional DAT channels internal to MXP. For increased bandwidth, this parameter need to be set to 2	RO	Configuration dependent
49	mpam_en	MPAM enable	RO	Configuration dependent
48	r2_enable	CMN R2 feature enable	RO	Configuration dependent
47	meta_data_en	Meta Data Preservation mode enable	RO	Configuration dependent
46:44	Reserved	Reserved	RO	-
43:36	rnsam_num_add_hashed_tgt	Number of additional hashed target ID's supported by the RN SAM, beyond the local HNF count	RO	Configuration dependent
35:32	num_remote_rnf	Number of remote RN-F devices in the system when the CML feature is enabled	RO	Configuration dependent

The following image shows the lower register bit assignments.

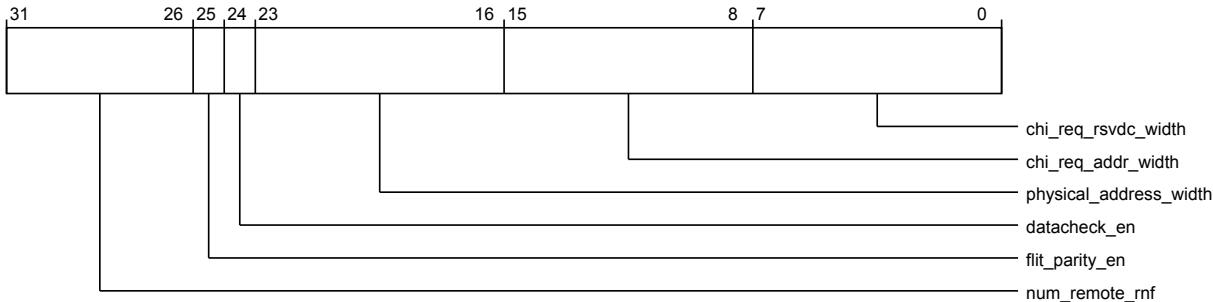


Figure 3-780 por_cfgm_por_info_global (low)

The following table shows the por_info_global lower register bit assignments.

Table 3-800 por_cfgm_por_info_global (low)

Bits	Field name	Description	Type	Reset
31:26	num_remote_rnf	Number of remote RN-F devices in the system when the CML feature is enabled	RO	Configuration dependent
25	flit_parity_en	Indicates whether parity checking is enabled in the transport layer on all flits sent on the interconnect	RO	Configuration dependent
24	datacheck_en	Indicates whether datacheck feature is enabled for CHI DAT flit	RO	Configuration dependent
23:16	physical_address_width	Physical address width	RO	Configuration dependent
15:8	chi_req_addr_width	REQ address width	RO	Configuration dependent
7:0	chi_req_rsvdc_width	RSVDC field width in CHI REQ flit	RO	Configuration dependent

por_info_global_1

Contains user-specified values of build-time global configuration parameters.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h908
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-781 por_cfgm_por_info_global_1 (high)

The following table shows the por_info_global_1 higher register bit assignments.

Table 3-801 por_cfgm_por_info_global_1 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

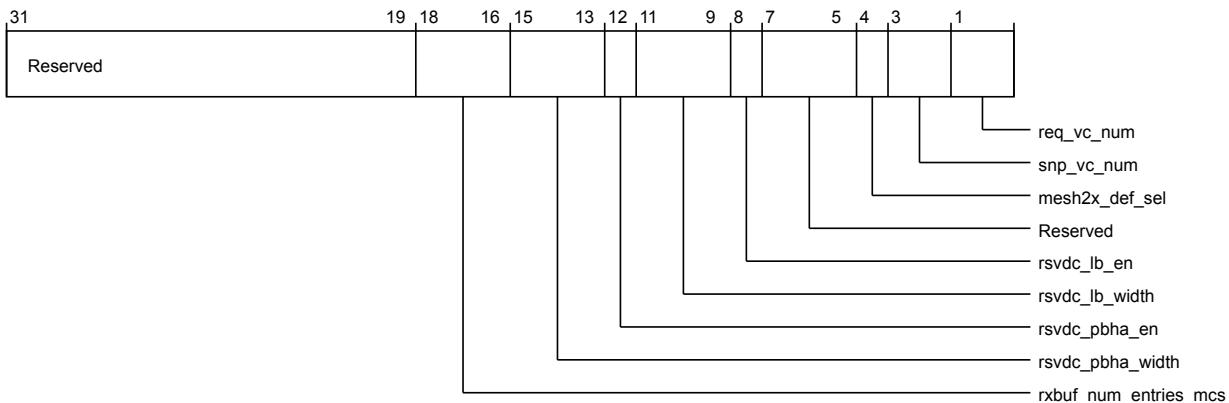


Figure 3-782 por_cfgm_por_info_global_1 (low)

The following table shows the por_info_global_1 lower register bit assignments.

Table 3-802 por_cfgm_por_info_global_1 (low)

Bits	Field name	Description	Type	Reset
31:19	Reserved	Reserved	RO	-
18:16	rxbuf_num_entries_mcs	RX Buffer Entries at upload interface of MCSX/MCSY	RO	Configuration dependent
15:13	rsvdc_pbha_width	RSVDC PBHA Field Width	RO	Configuration dependent
12	rsvdc_pbha_en	RSVDC PBHA Mode Enable	RO	Configuration dependent
11:9	rsvdc_lb_width	RSVDC Loop Back Field Width	RO	Configuration dependent
8	rsvdc_lb_en	RSVDC Loop Back Mode Enable	RO	Configuration dependent
7:5	Reserved	Reserved	RO	-
4	mesh2x_def_sel	Default ping-pong scheme selection for TGTID Lookup in 2xMESH	RO	Configuration dependent
3:2	snp_vc_num	Number of additional SNP channels internal to MXP. For increased bandwidth, this parameter need to be set to 2	RO	Configuration dependent
1:0	req_vc_num	Number of additional REQ channels internal to MXP. For increased bandwidth, this parameter need to be set to 2	RO	Configuration dependent

por_ppu_int_enable

Configures the HN-F PPU event interrupt. Contains the interrupt mask.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

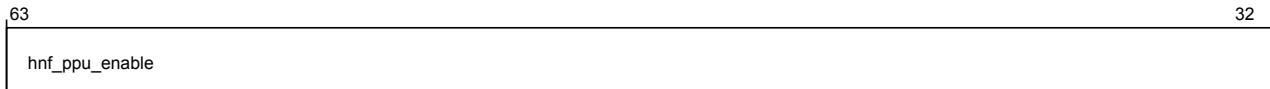


Figure 3-783 por_cfgm_por_ppu_int_enable (high)

The following table shows the por_ppu_int_enable higher register bit assignments.

Table 3-803 por_cfgm_por_ppu_int_enable (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_ppu_enable	Interrupt mask	RW	64'b0

The following image shows the lower register bit assignments.

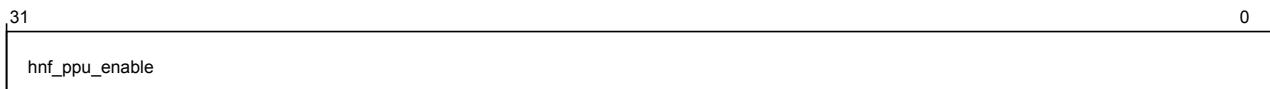


Figure 3-784 por_cfgm_por_ppu_int_enable (low)

The following table shows the por_ppu_int_enable lower register bit assignments.

Table 3-804 por_cfgm_por_ppu_int_enable (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_ppu_enable	Interrupt mask	RW	64'b0

por_ppu_int_enable_1

Configures the HN-F PPU event interrupt. Contains the interrupt mask.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C08
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63

32

hnf_ppu_enable_1

Figure 3-785 por_cfgm_por_ppu_int_enable_1 (high)

The following table shows the por_ppu_int_enable_1 higher register bit assignments.

Table 3-805 por_cfgm_por_ppu_int_enable_1 (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_ppu_enable_1	Interrupt mask	RW	64'b0

The following image shows the lower register bit assignments.

31

0

hnf_ppu_enable_1

Figure 3-786 por_cfgm_por_ppu_int_enable_1 (low)

The following table shows the por_ppu_int_enable_1 lower register bit assignments.

Table 3-806 por_cfgm_por_ppu_int_enable_1 (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_ppu_enable_1	Interrupt mask	RW	64'b0

por_ppu_int_status

Provides HN-F PPU event interrupt status.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h1C10

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63

32

hnf_ppu_status

Figure 3-787 por_cfgm_por_ppu_int_status (high)

The following table shows the por_ppu_int_status higher register bit assignments.

Table 3-807 por_cfgm_por_ppu_int_status (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_ppu_status	Interrupt status	W1C	64'b0

The following image shows the lower register bit assignments.

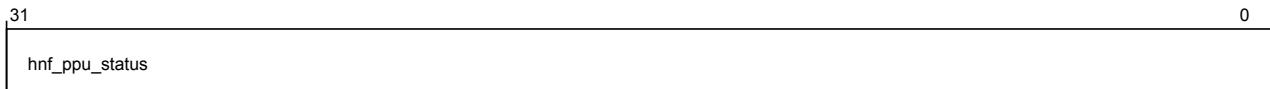


Figure 3-788 por_cfgm_por_ppu_int_status (low)

The following table shows the por_ppu_int_status lower register bit assignments.

Table 3-808 por_cfgm_por_ppu_int_status (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_ppu_status	Interrupt status	W1C	64'b0

por_ppu_int_status_1

Provides HN-F PPU event interrupt status.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h1C18

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

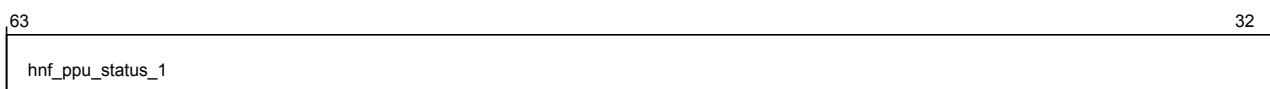


Figure 3-789 por_cfgm_por_ppu_int_status_1 (high)

The following table shows the por_ppu_int_status_1 higher register bit assignments.

Table 3-809 por_cfgm_por_ppu_int_status_1 (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_ppu_status_1	Interrupt status	W1C	64'b0

The following image shows the lower register bit assignments.



Figure 3-790 por_cfgm_por_ppu_int_status_1 (low)

The following table shows the por_ppu_int_status_1 lower register bit assignments.

Table 3-810 por_cfgm_por_ppu_int_status_1 (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_ppu_status_1	Interrupt status	W1C	64'b0

por_ppu_qactive_hyst

Number of hysteresis clock cycles to retain QACTIVE assertion

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1C20

Register reset 64'b00000000000000010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

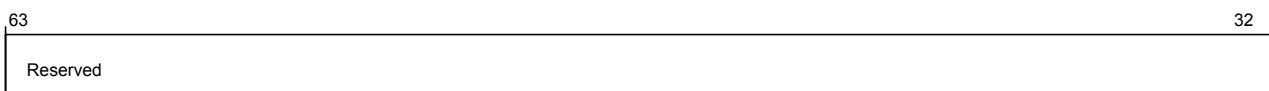


Figure 3-791 por_cfgm_por_ppu_qactive_hyst (high)

The following table shows the por_ppu_qactive_hyst higher register bit assignments.

Table 3-811 por_cfgm_por_ppu_qactive_hyst (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

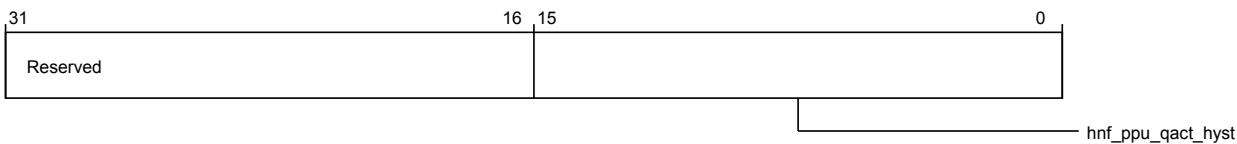


Figure 3-792 por_cfgm_por_ppu_qactive_hyst (low)

The following table shows the por_ppu_qactive_hyst lower register bit assignments.

Table 3-812 por_cfgm_por_ppu_qactive_hyst (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	hnf_ppu_qact_hyst	QACTIVE hysteresis	RW	16'h10

por_mpam_s_err_int_status

Provides HN-F MPAM Secure Error interrupt status.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h1C28

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-793 por_cfgm_por_mpam_s_err_int_status (high)

The following table shows the por_mpam_s_err_int_status higher register bit assignments.

Table 3-813 por_cfgm_por_mpam_s_err_int_status (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_mpam_s_err_int_status	MPAM S Interrupt status	W1C	64'b0

The following image shows the lower register bit assignments.

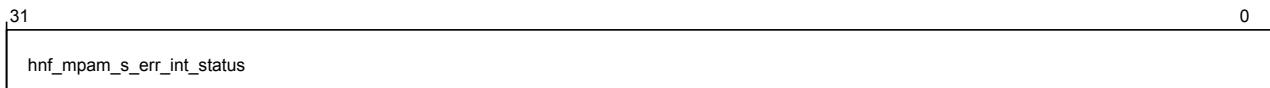


Figure 3-794 por_cfgm_por_mpam_s_err_int_status (low)

The following table shows the por_mpam_s_err_int_status lower register bit assignments.

Table 3-814 por_cfgm_por_mpam_s_err_int_status (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_mpam_s_err_int_status	MPAM S Interrupt status	W1C	64'b0

por_mpam_s_err_int_status_1

Provides HN-F MPAM Secure Error interrupt status.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h1C30
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

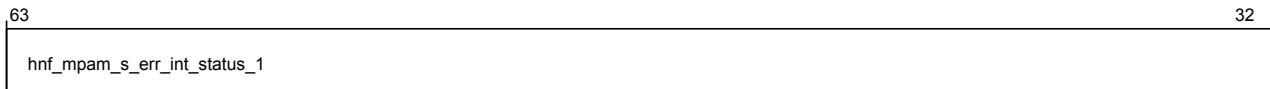


Figure 3-795 por_cfgm_por_mpam_s_err_int_status_1 (high)

The following table shows the por_mpam_s_err_int_status_1 higher register bit assignments.

Table 3-815 por_cfgm_por_mpam_s_err_int_status_1 (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_mpam_s_err_int_status_1	MPAM S Interrupt status	W1C	64'b0

The following image shows the lower register bit assignments.

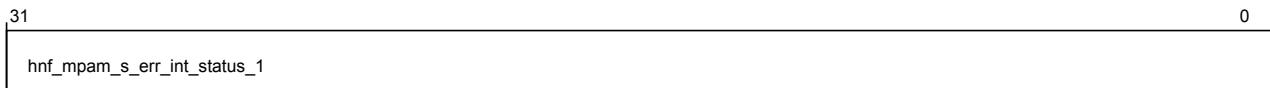


Figure 3-796 por_cfgm_por_mpam_s_err_int_status_1 (low)

The following table shows the por_mpam_s_err_int_status_1 lower register bit assignments.

Table 3-816 por_cfgm_por_mpam_s_err_int_status_1 (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_mpam_s_err_int_status_1	MPAM S Interrupt status	W1C	64'b0

por_mpam_ns_err_int_status

Provides HN-F MPAM Non-Secure Error interrupt status.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h1C38
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

hnf_mpam_ns_err_int_status

Figure 3-797 por_cfgm_por_mpam_ns_err_int_status (high)

The following table shows the por_mpam_ns_err_int_status higher register bit assignments.

Table 3-817 por_cfgm_por_mpam_ns_err_int_status (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_mpam_ns_err_int_status	MPAM NS Interrupt status	W1C	64'b0

The following image shows the lower register bit assignments.

31

0

hnf_mpam_ns_err_int_status

Figure 3-798 por_cfgm_por_mpam_ns_err_int_status (low)

The following table shows the por_mpam_ns_err_int_status lower register bit assignments.

Table 3-818 por_cfgm_por_mpam_ns_err_int_status (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_mpam_ns_err_int_status	MPAM NS Interrupt status	W1C	64'b0

por_mpam_ns_err_int_status_1

Provides HN-F MPAM Non-Secure Error interrupt status.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h1C40

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

32

hnf_mpam_ns_err_int_status_1

Figure 3-799 por_cfgm_por_mpam_ns_err_int_status_1 (high)

The following table shows the por_mpam_ns_err_int_status_1 higher register bit assignments.

Table 3-819 por_cfgm_por_mpam_ns_err_int_status_1 (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_mpam_ns_err_int_status_1	MPAM NS Interrupt status	W1C	64'b0

The following image shows the lower register bit assignments.

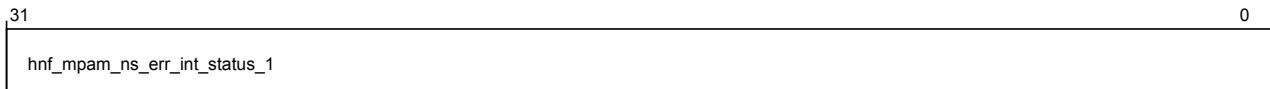


Figure 3-800 por_cfgm_por_mpam_ns_err_int_status_1 (low)

The following table shows the por_mpam_ns_err_int_status_1 lower register bit assignments.

Table 3-820 por_cfgm_por_mpam_ns_err_int_status_1 (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_mpam_ns_err_int_status_1	MPAM NS Interrupt status	W1C	64'b0

por_cfgm_child_pointer_0-255

This register repeats 255 times. It parametrized by the index from 0 to 255. Contains base address of child configuration node. NOTE: There will be as many child pointer registers in the Global Config Unit as the number of XP's on the chip. Each successive child pointer register will be at the next 8 byte address boundary. Each successive child pointer register will be named with the suffix corresponding to the register number. For example por_cfgm_child_pointer_<0:255>

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h100 + # {8 * [0, 1, 2, .., 254, 255]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

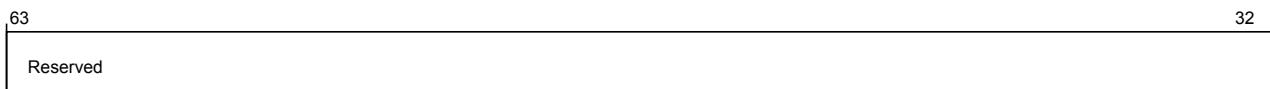


Figure 3-801 por_cfgm_por_cfgm_child_pointer_0-255 (high)

The following table shows the por_cfgm_child_pointer_0-255 higher register bit assignments.

Table 3-821 por_cfgm_por_cfgm_child_pointer_0-255 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

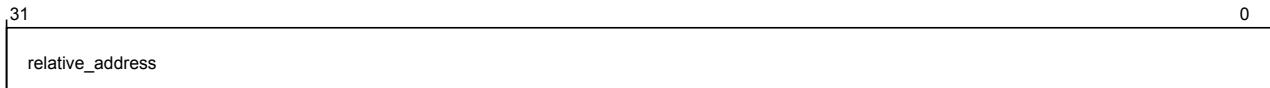


Figure 3-802 por_cfgm_por_cfgm_child_pointer_0-255 (low)

The following table shows the por_cfgm_por_cfgm_child_pointer_0-255 lower register bit assignments.

Table 3-822 por_cfgm_por_cfgm_child_pointer_0-255 (low)

Bits	Field name	Description	Type	Reset
31:0	relative_address	Bit 31: External or internal child node 1'b1: Indicates child pointer points to a configuration node that is external to CMN-600 1'b0: Indicates child pointer points to a configuration node that is internal to CMN-600 Bits [30]: Set to 1'b0 Bits [29:0]: Child node address offset relative to PERIPHBASE	RO	32'b0

3.3.13 CCG_RA register descriptions

This section lists the HN-F registers.

por_ccg_ra_node_info

Provides component identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h0
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

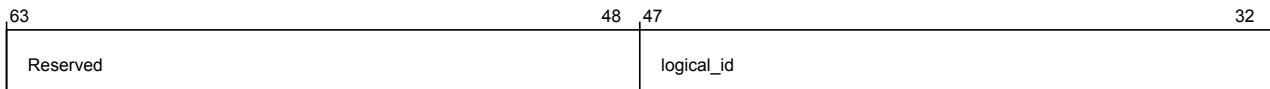


Figure 3-803 por_ccg_ra_por_ccg_ra_node_info (high)

The following table shows the por_ccg_ra_node_info higher register bit assignments.

Table 3-823 por_ccg_ra_por_ccg_ra_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

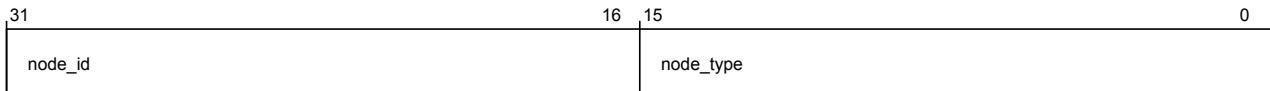


Figure 3-804 por_ccg_ra_por_ccg_ra_node_info (low)

The following table shows the por_ccg_ra_node_info lower register bit assignments.

Table 3-824 por_ccg_ra_por_ccg_ra_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0103

por_ccg_ra_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-805 por_ccg_ra_por_ccg_ra_child_info (high)

The following table shows the por_ccg_ra_child_info higher register bit assignments.

Table 3-825 por_ccg_ra_por_ccg_ra_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

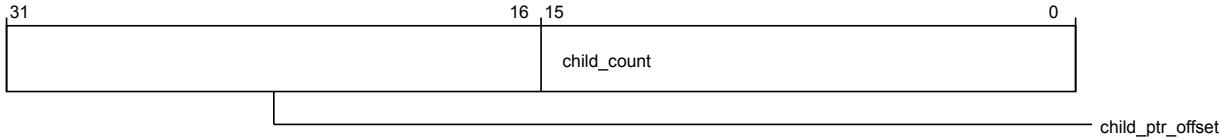


Figure 3-806 por_ccg_ra_por_ccg_ra_child_info (low)

The following table shows the por_ccg_ra_child_info lower register bit assignments.

Table 3-826 por_ccg_ra_por_ccg_ra_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'h0

por_ccg_ra_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

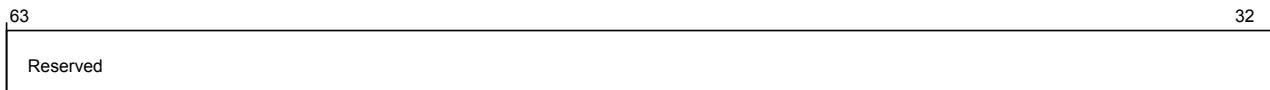


Figure 3-807 por_ccg_ra_por_ccg_ra_secure_register_groups_override (high)

The following table shows the por_ccg_ra_secure_register_groups_override higher register bit assignments.

Table 3-827 por_ccg_ra_por_ccg_ra_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

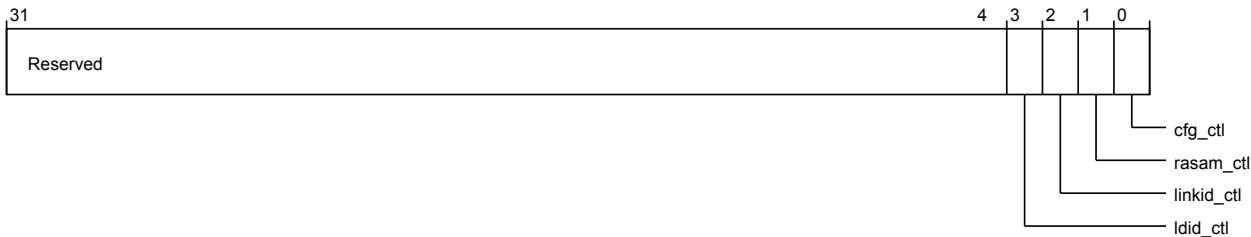


Figure 3-808 por_ccg_ra_por_ccg_ra_secure_register_groups_override (low)

The following table shows the por_ccg_ra_secure_register_groups_override lower register bit assignments.

Table 3-828 por_ccg_ra_por_ccg_ra_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	ldid_ctl	Allows non-secure access to secure RA LDID registers	RW	1'b0
2	linkid_ctl	Allows non-secure access to secure RA Link ID registers	RW	1'b0
1	rasam_ctl	Allows non-secure access to secure RA SAM control registers	RW	1'b0
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_ccg_ra_unit_info

Provides component identification information for CXRA.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

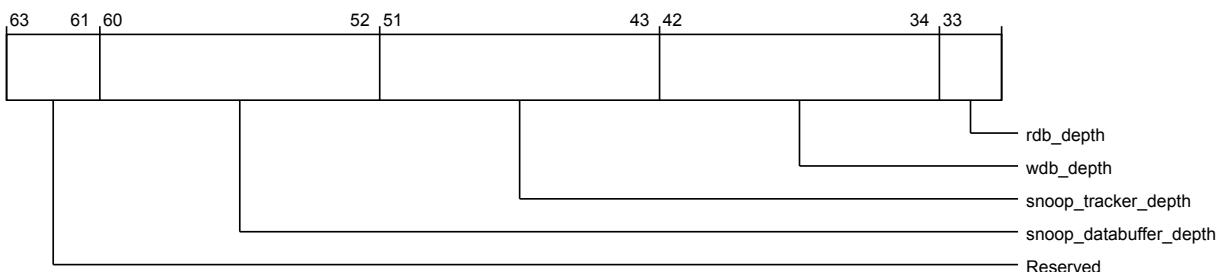


Figure 3-809 por_ccg_ra_por_ccg_ra_unit_info (high)

The following table shows the por_ccg_ra_unit_info higher register bit assignments.

Table 3-829 por_ccg_ra_por_ccg_ra_unit_info (high)

Bits	Field name	Description	Type	Reset
63:61	Reserved	Reserved	RO	-
60:52	snoop_databuffer_depth	Depth of Snoop Data Buffer - number of outstanding SNP requests on CHI	RO	Configuration dependent
51:43	snoop_tracker_depth	Depth of Snoop Tracker - number of outstanding SNP requests on CCIX	RO	Configuration dependent
42:34	wdb_depth	Depth of Write Data Buffer	RO	Configuration dependent
33:32	rdb_depth	Depth of Read Data Buffer	RO	Configuration dependent

The following image shows the lower register bit assignments.

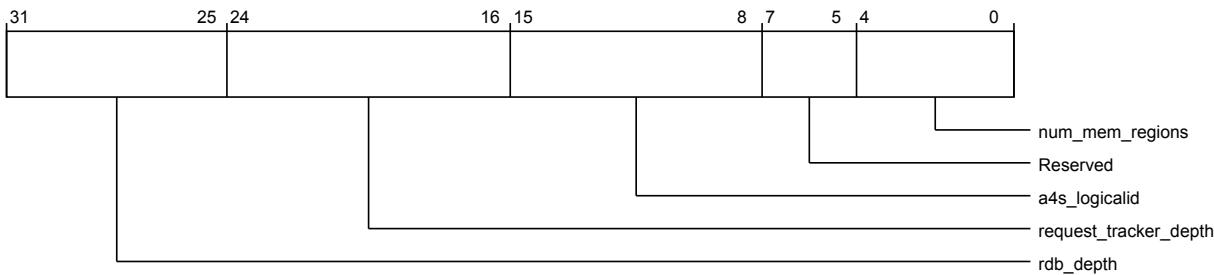


Figure 3-810 por_ccg_ra_por_ccg_ra_unit_info (low)

The following table shows the por_ccg_ra_unit_info lower register bit assignments.

Table 3-830 por_ccg_ra_por_ccg_ra_unit_info (low)

Bits	Field name	Description	Type	Reset
31:25	rdb_depth	Depth of Read Data Buffer	RO	Configuration dependent
24:16	request_tracker_depth	Depth of Request Tracker - number of outstanding Memory requests on CCIX	RO	Configuration dependent
15:8	a4s_logicalid	AXI4Stream interfaces logical ID	RO	Configuration dependent
7:5	Reserved	Reserved	RO	-
4:0	num_mem_regions	Number of memory regions supported	RO	Configuration dependent

por_ccg_ra_cfg_ctl

Functions as the configuration control register. Specifies the current mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b00111

Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_ccg_ra_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

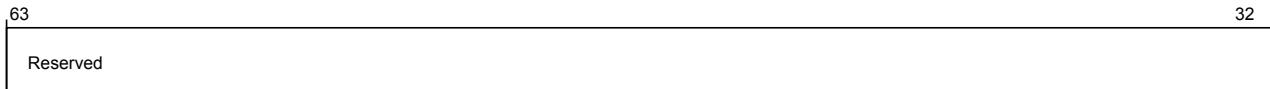


Figure 3-811 por_ccg_ra_por_ccg_ra_cfg_ctl (high)

The following table shows the por_ccg_ra_cfg_ctl higher register bit assignments.

Table 3-831 por_ccg_ra_por_ccg_ra_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

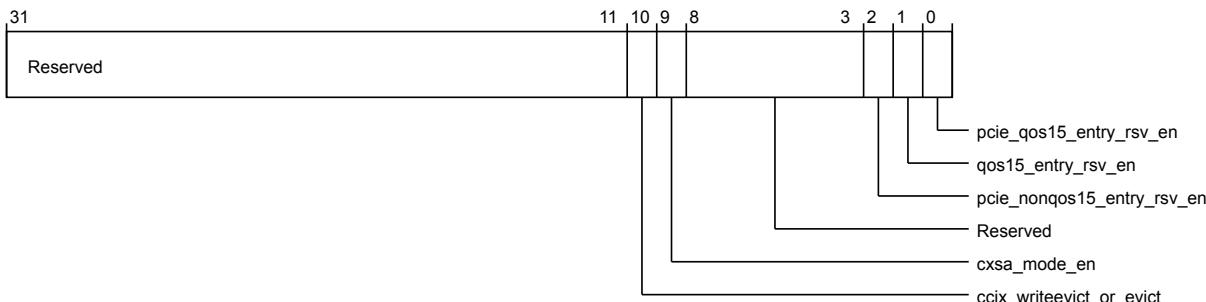


Figure 3-812 por_ccg_ra_por_ccg_ra_cfg_ctl (low)

The following table shows the por_ccg_ra_cfg_ctl lower register bit assignments.

Table 3-832 por_ccg_ra_por_ccg_ra_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10	ccix_writeevict_or_evict	When set, downgrades WriteEvictOrEvict to Evict 1'b1: Evict is sent 1'b0: WriteEvict is sent	RW	1'b0
9	cxsa_mode_en	When set, enables the CCIX Slave Agent mode. In this mode RA functions as a CCIX Slave Agent 1'b1: CCIX Slave Agent 1'b0: CCIX Requesting Agent	RW	1'b0
8:3	Reserved	Reserved	RO	-

Table 3-832 por_ccg_ra_por_ccg_ra_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	pcie_nonqos15_entry_rsv_en	Enables entry reservation for non QoS15 traffic from PCIe RN-I/RN-D 1'b1: Reserves tracker entry for non QoS15 requests from PCIe RN-I/RN-D 1'b0: Does not reserve tracker entry for non QoS15 requests from PCIe RN-I/RN-D	RW	1'b1
1	qos15_entry_rsv_en	Enables entry reservation for QoS15 traffic 1'b1: Reserves tracker entry for QoS15 requests 1'b0: Does not reserve tracker entry for QoS15 requests	RW	1'b1
0	pcie_qos15_entry_rsv_en	Enables entry reservation for QoS15 traffic from PCIe RN-I/RN-D 1'b1: Reserves tracker entry for QoS15 requests from PCIe RN-I/RN-D 1'b0: Does not reserve tracker entry for QoS15 requests from PCIe RN-I/RN-D	RW	1'b1

por_ccg_ra_aux_ctl

Functions as the auxiliary control register for CXRA.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	64'b0011110011100011100001000110
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

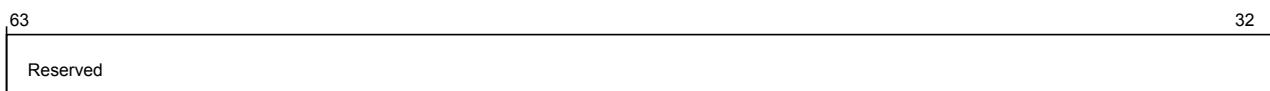


Figure 3-813 por_ccg_ra_por_ccg_ra_aux_ctl (high)

The following table shows the por_ccg_ra_aux_ctl higher register bit assignments.

Table 3-833 por_ccg_ra_por_ccg_ra_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

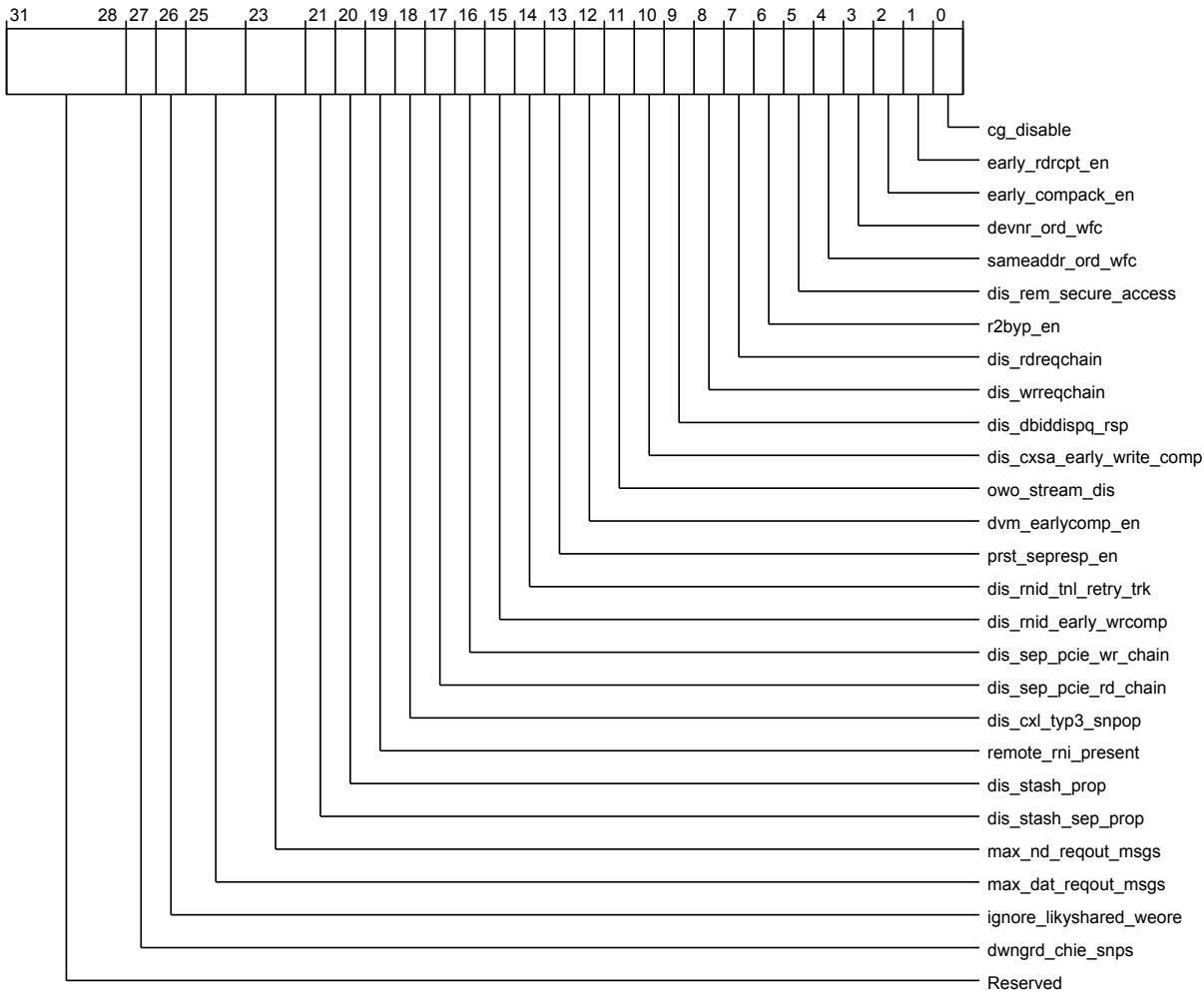


Figure 3-814 por_ccg_ra_por_ccg_ra_aux_ctl (low)

The following table shows the por_ccg_ra_aux_ctl lower register bit assignments.

Table 3-834 por_ccg_ra_por_ccg_ra_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27	dwngrd_chie_snps	When set, downgrades CHIE SnpPreferUnique to SnpNotSharedDirty	RW	1'b0
26	ignore_likyshared_weore	When set, disables the use of LikelyShared(LS) bit to make a decision for WriteEvictOrEvict 1'b0: Send WriteEvict when LS= 0 and send Evict when LS=1 1'b1: Ignore LS bit. WriteEvict is sent. Further static decision can be made using ccix_writeevict_or_evict in cfg_ctl register	RW	1'b0

Table 3-834 por_ccg_ra_por_ccg_ra_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
25:24	max_dat_reqout_msgs	<p>Used to configure the maximum number of data requests messages (writes, atomics etc.) presented to CCLA's packing logic.</p> <p>2'b00: one message 2'b01: two messages 2'b10: three messages 2'b11: four messages</p> <p>Note: The max is further limited by max allowed by the given protocol (CCIX2.0/CXL.mem/CXL.cache)</p>	RW	2'b11
23:22	max_nd_reqout_msgs	<p>Used to configure the maximum number of non-data requests messages (reads, dataless) presented to CCLA's packing logic.</p> <p>2'b00: one message 2'b01: two messages 2'b10: three messages 2'b11: four messages</p> <p>Note: The max is further limited by max allowed by the given protocol (CCIX2.0/CXL.mem/CXL.cache)</p>	RW	2'b11
21	dis_stash_sep_prop	When set, disables propagation of StashSep opcodes on CCIX. StashSep opcodes are sent as Stash opcodes when set. Applicable only in SMP mode	RW	1'b0
20	dis_stash_prop	When set, disables propagation of stash opcodes on CCIX. Applicable only in SMP mode	RW	1'b0
19	remote_rni_present	When set, Enables TXNID coloring to enable traffic to remote RNI	RW	1'b1
18	dis_cxl_typ3_snpop	When set, drives SnpType= NOP on CXL Type3 M2S Req and Rwd messages	RW	1'b1
17	dis_sep_PCIE_rd_chain	When set, disables separate ordering chains for PCIe reads. By default, PCIe read chain is disabled	RW	1'b1
16	dis_sep_PCIE_wr_chain	When set, disables separate ordering chains for PCIe writes. By default, PCIe write chain is enabled	RW	1'b0
15	dis_rnid_early_wrcomp	When set, disables early write completions for tunneled writes from RNI.	RW	1'b0
14	dis_rnid_tnl_retry_trk	When set, disables RNID write request tunneling retry tracker.	RW	1'b0
13	prst_sepresp_en	<p>When set, enables separate persist response on CCIX for persistent cache maintenance (PCMO2) operation</p> <p>Note: this bit is applicable only in SMP mode.</p>	RW	1'b1
12	dvm_earlycomp_en	When set, enables early DVM Op completion responses from RA.	RW	1'b1
11	owo_stream_dis	When set, disables CompAck dependency to dispatch an ordered PCIe write.	RW	1'b1
10	dis_cxsas_early_write_comp	When set, disables early write completions in CCIX Slave Agent mode.	RW	1'b0

Table 3-834 por_ccg_ra_por_ccg_ra_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
9	dis_dbiddispq_rsp	When set, disables the dispatch of DBID responses from a separate DispatchQ.	RW	1'b0
8	dis_wreqchain	When set, disables chaining of write requests.	RW	1'b0
7	dis_rdreqchain	When set, disables chaining of read and dataless requests.	RW	1'b0
6	r2byp_en	When set, enables request bypass. Applies to read and dataless requests only. Note: When set will affect the capability to chain a request on the TX side	RW	1'b1
5	dis_rem_secure_access	When set, treats all the incoming snoops as non-secure and forces the NS bit to 1	RW	1'b0
4	sameaddr_ord_wfc	When set, enables waiting for completion (COMP) before dispatching next same Addr dependent transaction (TXN)	RW	1'b0
3	devnr_ord_wfc	When set, enables waiting for completion (COMP) before dispatching next Device-nR dependent transaction (TXN)	RW	1'b0
2	early_compack_en	Early CompAck enable; enables sending early CompAck on CCIX for requests that require CompAck	RW	1'b1
1	early_rdrcpt_en	Early ReadReceipt enable; enables sending early ReadReceipt for ordered read requests	RW	1'b1
0	cg_disable	Disables clock gating when set	RW	1'b0

por_ccg_ra_cbusy_limit_ctl

Cbusy threshold limits for RHT entries.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA18
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

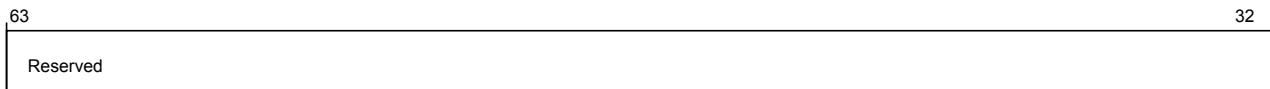


Figure 3-815 por_ccg_ra_por_ccg_ra_cbusy_limit_ctl (high)

The following table shows the por_ccg_ra_cbusy_limit_ctl higher register bit assignments.

Table 3-835 por_ccg_ra_por_ccg_ra_cbusy_limit_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

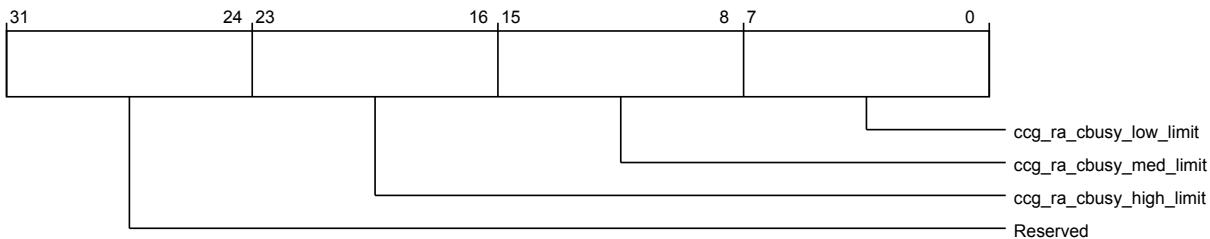


Figure 3-816 por_ccg_ra_por_ccg_ra_cbusy_limit_ctl (low)

The following table shows the por_ccg_ra_cbusy_limit_ctl lower register bit assignments.

Table 3-836 por_ccg_ra_por_ccg_ra_cbusy_limit_ctl (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	ccg_ra_cbusy_high_limit	RHT limit for CBusy High	RW	Configuration dependent
15:8	ccg_ra_cbusy_med_limit	RHT limit for CBusy Med	RW	Configuration dependent
7:0	ccg_ra_cbusy_low_limit	RHT limit for CBusy Low	RW	Configuration dependent

por_ccg_ra_sam_addr_region_reg_0-7

This register repeats 7 times. It parametrized by the i from 0 to 7. Configures Address Region #*{i}* for RA SAM.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC00 + #*{8*[0, 1, 2, 3, 4, 5, 6, 7]}*

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ra_secure_register_groups_override.rasam_ctl

The following image shows the higher register bit assignments.

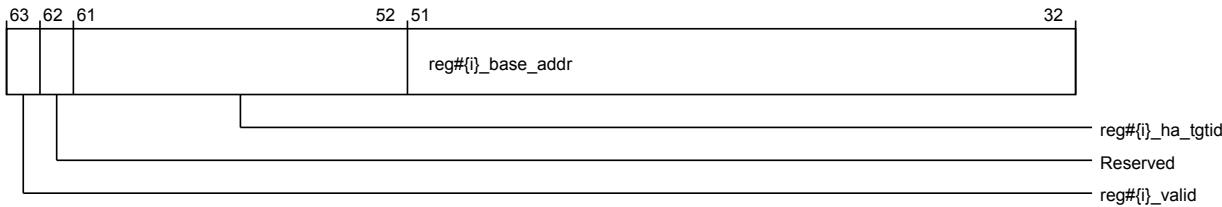


Figure 3-817 por_ccg_ra_por_ccg_ra_sam_addr_region_reg_0-7 (high)

The following table shows the por_ccg_ra_sam_addr_region_reg_0-7 higher register bit assignments.

Table 3-837 por_ccg_ra_por_ccg_ra_sam_addr_region_reg_0-7 (high)

Bits	Field name	Description	Type	Reset
63	reg#{i}_valid	Specifies if the memory region is valid	RW	1'b0
62	Reserved	Reserved	RO	-
61:52	reg#{i}_ha_tgtid	Specifies the target HAID	RW	10'b0
51:32	reg#{i}_base_addr	Specifies the 2^n-aligned base address for the memory region	RW	36'h0

The following image shows the lower register bit assignments.

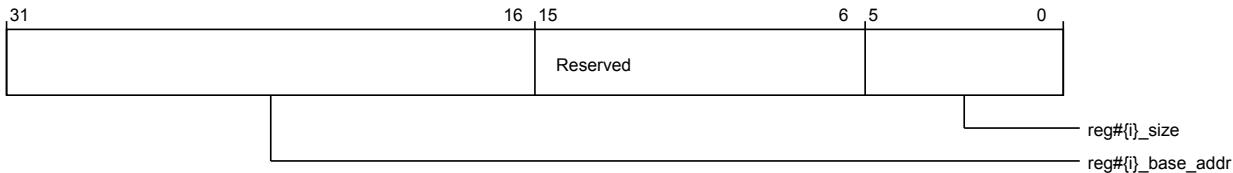


Figure 3-818 por_ccg_ra_por_ccg_ra_sam_addr_region_reg_0-7 (low)

The following table shows the por_ccg_ra_sam_addr_region_reg_0-7 lower register bit assignments.

Table 3-838 por_ccg_ra_por_ccg_ra_sam_addr_region_reg_0-7 (low)

Bits	Field name	Description	Type	Reset
31:16	reg#{i}_base_addr	Specifies the 2^n-aligned base address for the memory region	RW	36'h0
15:6	Reserved	Reserved	RO	-
5:0	reg#{i}_size	Specifies the size of the memory region	RW	1'b0

por_ccg_ra_agentid_to_linkid_val

Specifies which Agent ID to Link ID mappings are valid.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD00

Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ra_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

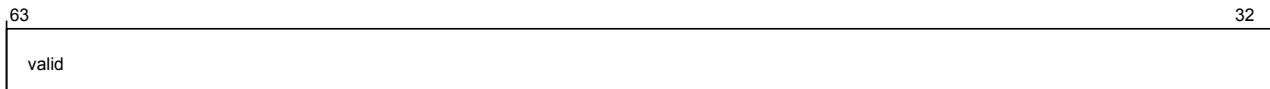


Figure 3-819 por_ccg_ra_por_ccg_ra_agentid_to_linkid_val (high)

The following table shows the por_ccg_ra_agentid_to_linkid_val higher register bit assignments.

Table 3-839 por_ccg_ra_por_ccg_ra_agentid_to_linkid_val (high)

Bits	Field name	Description	Type	Reset
63:32	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

The following image shows the lower register bit assignments.

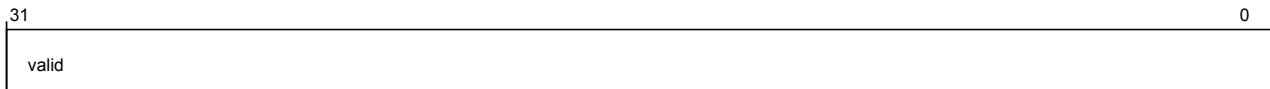


Figure 3-820 por_ccg_ra_por_ccg_ra_agentid_to_linkid_val (low)

The following table shows the por_ccg_ra_agentid_to_linkid_val lower register bit assignments.

Table 3-840 por_ccg_ra_por_ccg_ra_agentid_to_linkid_val (low)

Bits	Field name	Description	Type	Reset
31:0	valid	Specifies whether the Link ID is valid; bit number corresponds to logical Agent ID number (from 0 to 63)	RW	63'h0

por_ccg_ra_agentid_to_linkid_reg_0-7

This register repeats 7 times. It parametrized by the i from 0 to 7. Specifies the mapping of Agent ID to Link ID for Agent IDs # $\{i*8\}$ to # $\{i*8+7\}$.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD10 + # $\{[0, 1, 2, 3, 4, 5, 6, 7]*8\}$
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ra_secure_register_groups_override.linkid_ctl

The following image shows the higher register bit assignments.

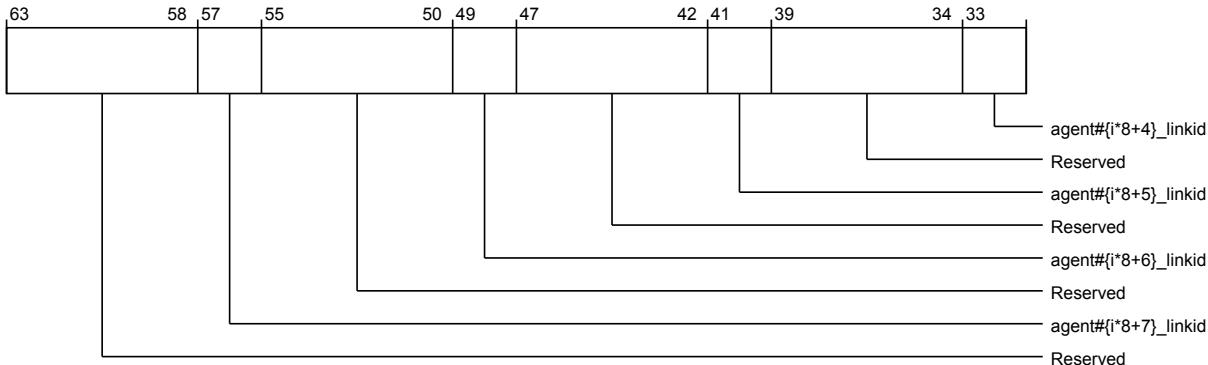


Figure 3-821 por_ccg_ra_por_ccg_ra_agentid_to_linkid_reg_0-7 (high)

The following table shows the por_ccg_ra_agentid_to_linkid_reg_0-7 higher register bit assignments.

Table 3-841 por_ccg_ra_por_ccg_ra_agentid_to_linkid_reg_0-7 (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:56	agent#{i*8+7}_linkid	Specifies the Link ID for Agent ID #{i*8+7}	RW	2'h0
55:50	Reserved	Reserved	RO	-
49:48	agent#{i*8+6}_linkid	Specifies the Link ID for Agent ID #{i*8+6}	RW	2'h0
47:42	Reserved	Reserved	RO	-
41:40	agent#{i*8+5}_linkid	Specifies the Link ID for Agent ID #{i*8+5}	RW	2'h0
39:34	Reserved	Reserved	RO	-
33:32	agent#{i*8+4}_linkid	Specifies the Link ID for Agent ID #{i*8+4}	RW	2'h0

The following image shows the lower register bit assignments.

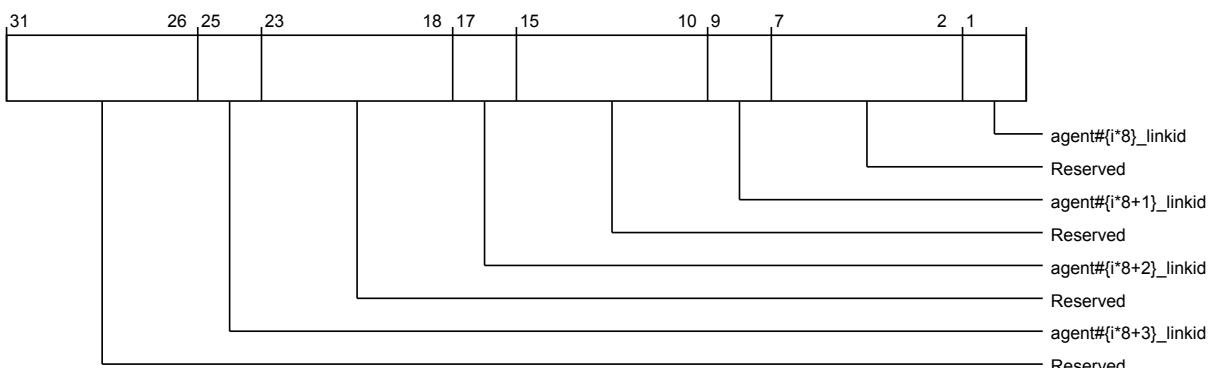


Figure 3-822 por_ccg_ra_por_ccg_ra_agentid_to_linkid_reg_0-7 (low)

The following table shows the por_ccg_ra_agentid_to_linkid_reg_0-7 lower register bit assignments.

Table 3-842 por_ccg_ra_por_ccg_ra_agentid_to_linkid_reg_0-7 (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	agent#{i*8+3}_linkid	Specifies the Link ID for Agent ID #{i*8+3}	RW	2'h0
23:18	Reserved	Reserved	RO	-
17:16	agent#{i*8+2}_linkid	Specifies the Link ID for Agent ID #{i*8+2}	RW	2'h0
15:10	Reserved	Reserved	RO	-
9:8	agent#{i*8+1}_linkid	Specifies the Link ID for Agent ID #{i*8+1}	RW	2'h0
7:2	Reserved	Reserved	RO	-
1:0	agent#{i*8}_linkid	Specifies the Link ID for Agent ID #{i*8}	RW	2'h0

por_ccg_ra_rni_lidid_to_exp_raid_reg_0-9

This register repeats 9 times. It parametrized by the i from 0 to 9. Specifies the mapping of RN-I's LDID to Expanded RAID for LDIDs # $\{i^*4\}$ to # $\{i^*4+3\}$.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hE

Register Reset 64 b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_ccg_ra_secure_register_groups_override.ldid_ctl

The following section details the high-level architecture of the system.

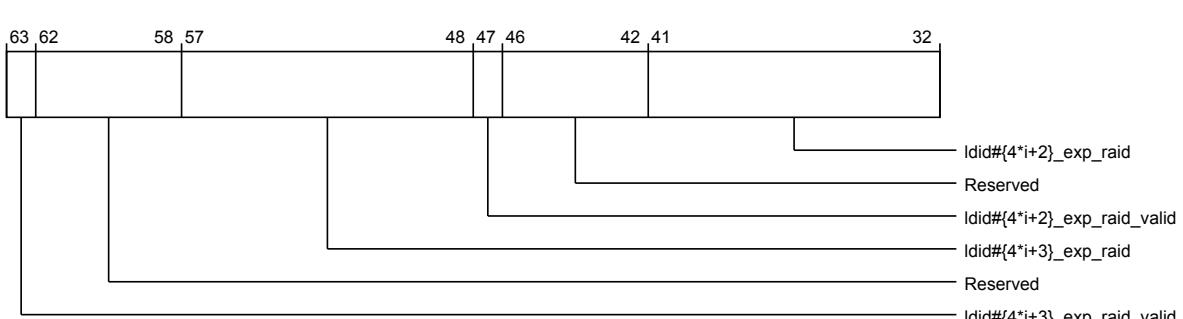


Figure 3-823 por ccq ra por ccq ra rni ldid to exp raid req 0-9 (high)

The following table shows the port configuration assignments for RAID reg 0-9 higher register bit assignments.

Table 3-843 por_ccg_ra_por_ccg_ra_rni_ldid_to_exp_raid_reg_0-9 (high)

Bits	Field name	Description	Type	Reset
63	ldid#{4*i+3}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i+3} is valid;	RW	1'h0
62:58	Reserved	Reserved	RO	-
57:48	ldid#{4*i+3}_exp_raid	Specifies the Expanded RAID for LDID #{4*i+3}	RW	10'h0
47	ldid#{4*i+2}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i+2} is valid;	RW	1'h0
46:42	Reserved	Reserved	RO	-
41:32	ldid#{4*i+2}_exp_raid	Specifies the Expanded RAID for LDID #{4*i+2}	RW	10'h0

The following image shows the lower register bit assignments.

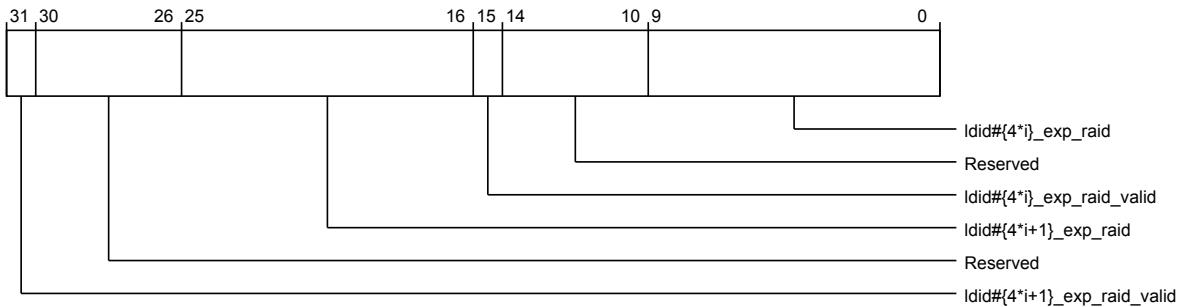


Figure 3-824 por_ccg_ra_por_ccg_ra_rni_ldid_to_exp_raid_reg_0-9 (low)

The following table shows the por_ccg_ra_rni_ldid_to_exp_raid_reg_0-9 lower register bit assignments.

Table 3-844 por_ccg_ra_por_ccg_ra_rni_ldid_to_exp_raid_reg_0-9 (low)

Bits	Field name	Description	Type	Reset
31	ldid#{4*i+1}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i+1} is valid;	RW	1'h0
30:26	Reserved	Reserved	RO	-
25:16	ldid#{4*i+1}_exp_raid	Specifies the Expanded RAID for LDID #{4*i+1}	RW	10'h0
15	ldid#{4*i}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{4*i} is valid;	RW	1'h0
14:10	Reserved	Reserved	RO	-
9:0	ldid#{4*i}_exp_raid	Specifies the Expanded RAID for LDID #{4*i}	RW	10'h0

por_ccg_ra_rnd_ldid_to_exp_raid_reg_0-9

This register repeats 9 times. It parametrized by the i from 0 to 9. Specifies the mapping of RN-D's LDID to Expanded RAID for LDIDs #{{i}*4} to #{{i}*4+3}.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset	16'hF00 + #[{0, 1, 2, .., 8, 9}*8]
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

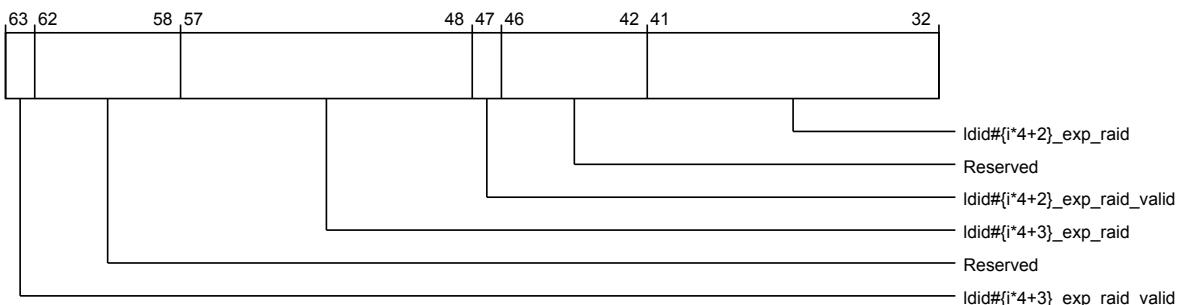


Figure 3-825 por_ccg_ra_por_ccg_ra_rnd_Idid_to_exp_raid_reg_0-9 (high)

The following table shows the por_ccg_ra_rnd_ldid_to_exp_raid_reg_0-9 higher register bit assignments.

Table 3-845 por_ccg_ra_por_ccg_ra_rnd_ldid_to_exp_raid_reg_0-9 (high)

Bits	Field name	Description	Type	Reset
63	ldid#{i*4+3}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4+3} is valid;	RW	1'h0
62:58	Reserved	Reserved	RO	-
57:48	ldid#{i*4+3}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+3}	RW	10'h0
47	ldid#{i*4+2}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4+2} is valid;	RW	1'h0
46:42	Reserved	Reserved	RO	-
41:32	ldid#{i*4+2}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+2}	RW	10'h0

The following image shows the lower register bit assignments.

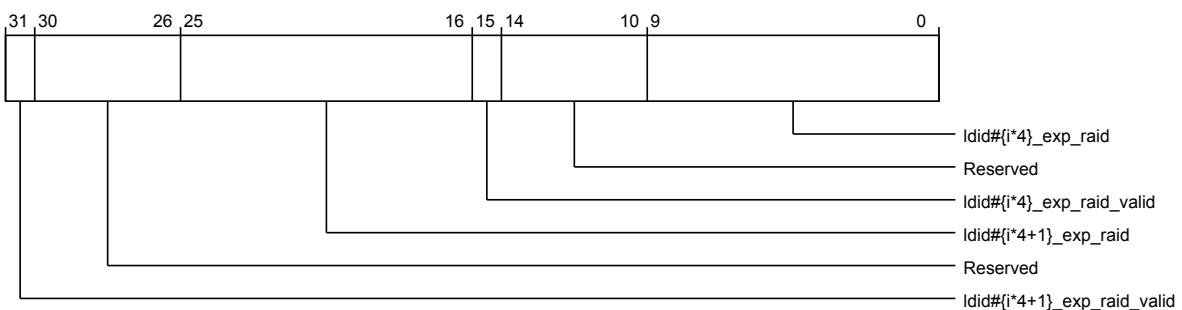


Figure 3-826 por_ccg_ra_por_ccg_ra_rnd_ldid_to_exp_raid_reg_0-9 (low)

The following table shows the port configuration register assignments for RAID 0-9 lower register bit assignments.

Table 3-846 por_ccg_ra_por_ccg_ra_rnd_ldid_to_exp_raid_reg_0-9 (low)

Bits	Field name	Description	Type	Reset
31	ldid#{i*4+1}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4+1} is valid;	RW	1'h0
30:26	Reserved	Reserved	RO	-
25:16	ldid#{i*4+1}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+1}	RW	10'h0
15	ldid#{i*4}_exp_raid_valid	Specifies whether the Expanded RAID for LDID#{i*4} is valid;	RW	1'h0
14:10	Reserved	Reserved	RO	-
9:0	ldid#{i*4}_exp_raid	Specifies the Expanded RAID for LDID #{i*4}	RW	10'h0

por_ccg_ra_rnf_lidid_to_exp_raid_reg_0-127

This register repeats 127 times. It parametrized by the i from 0 to 127. Specifies the mapping of RN-F's LDID to Expanded RAID for LDIDs # $\{i^*4\}$ to # $\{i^*4+3\}$.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1000 + {[0, 1, 2, .., 126, 127]*8}

Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

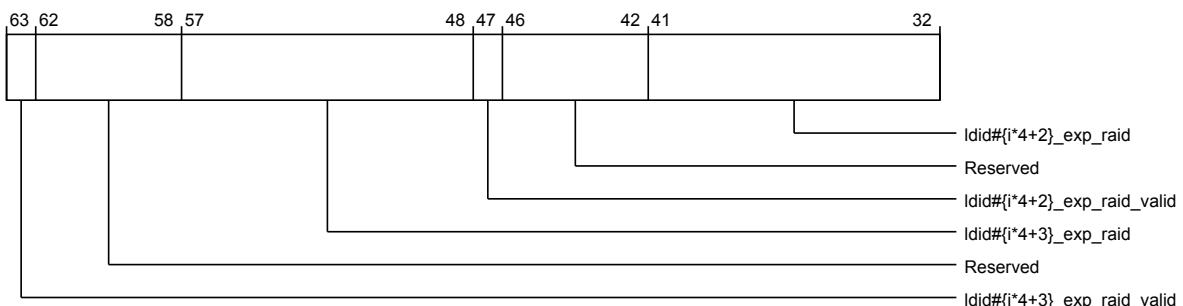


Figure 3-827 por_ccg_ra por_ccg_ra rnf_ldid_to_exp_raid_reg_0-127 (high)

The following table shows the por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127 higher register bit assignments.

Table 3-847 por_ccg_ra_por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127 (high)

Bits	Field name	Description	Type	Reset
63	ldid#{i*4+3}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4+3} is valid;	RW	1'h0
62:58	Reserved	Reserved	RO	-

Table 3-847 por_ccg_ra_por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127 (high) (continued)

Bits	Field name	Description	Type	Reset
57:48	ldid#{i*4+3}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+3}	RW	10'h0
47	ldid#{i*4+2}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4+2} is valid;	RW	1'h0
46:42	Reserved	Reserved	RO	-
41:32	ldid#{i*4+2}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+2}	RW	10'h0

The following image shows the lower register bit assignments.

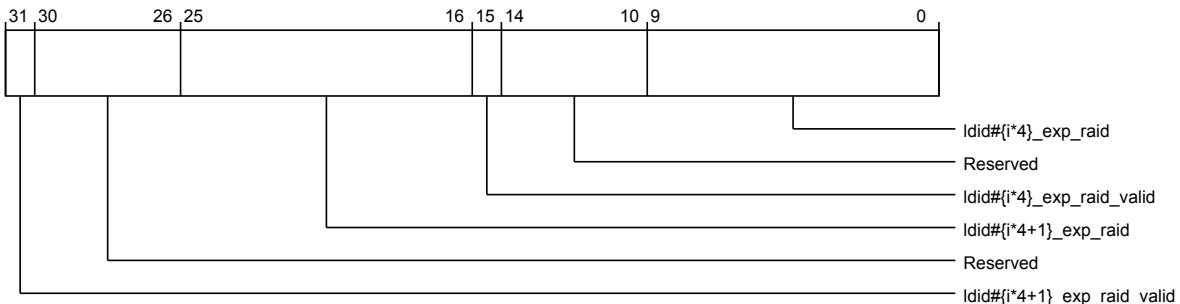


Figure 3-828 por_ccg_ra_por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127 (low)

The following table shows the por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127 lower register bit assignments.

Table 3-848 por_ccg_ra_por_ccg_ra_rnf_ldid_to_exp_raid_reg_0-127 (low)

Bits	Field name	Description	Type	Reset
31	ldid#{i*4+1}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4+1} is valid;	RW	1'h0
30:26	Reserved	Reserved	RO	-
25:16	ldid#{i*4+1}_exp_raid	Specifies the Expanded RAID for LDID #{i*4+1}	RW	10'h0
15	ldid#{i*4}_exp_raid_valid	Specifies whether the look table entry for default LDID#{i*4} is valid;	RW	1'h0
14:10	Reserved	Reserved	RO	-
9:0	ldid#{i*4}_exp_raid	Specifies the Expanded RAID for LDID #{i*4}	RW	10'h0

por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127

This register repeats 127 times. It parametrized by the i from 0 to 127. Specifies the mapping of RN-F's Default LDID to CHI NodeID for LDIDs #{i*4} to #{i*4+3}.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1400 + #[0, 1, 2, .., 126, 127]*8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_ccg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

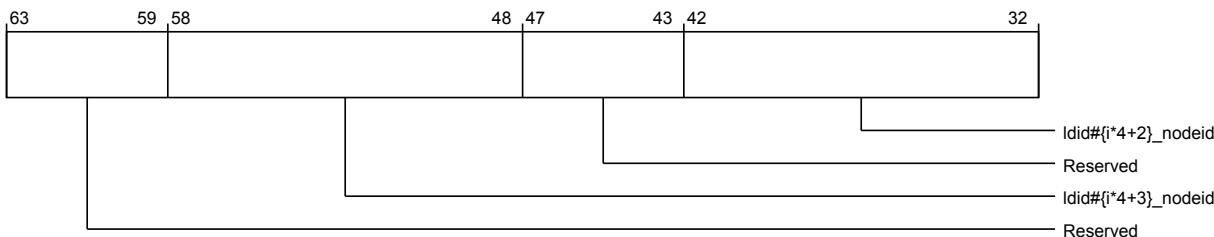


Figure 3-829 por_ccg_ra_por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127 (high)

The following table shows the por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127 higher register bit assignments.

Table 3-849 por_ccg_ra_por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	ldid#{i*4+3}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4+3\}$	RO	11'h0
47:43	Reserved	Reserved	RO	-
42:32	ldid#{i*4+2}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4+2\}$	RO	11'h0

The following image shows the lower register bit assignments.

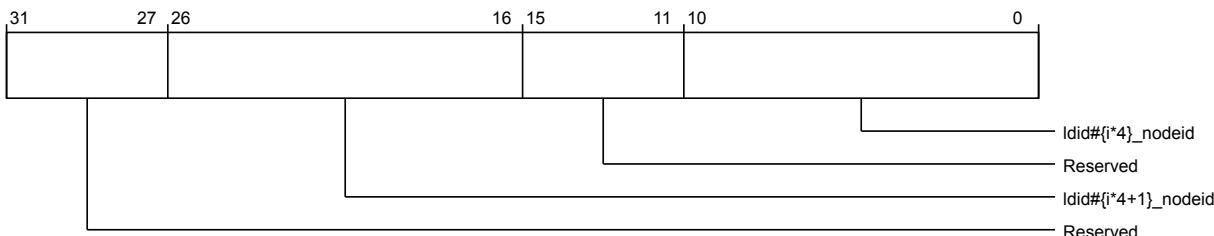


Figure 3-830 por_ccg_ra_por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127 (low)

The following table shows the por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127 lower register bit assignments.

Table 3-850 por_ccg_ra_por_ccg_ra_rnf_ldid_to_nodeid_reg_0-127 (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:16	ldid#{i*4+1}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4+1\}$	RO	11'h0
15:11	Reserved	Reserved	RO	-
10:0	ldid#{i*4}_nodeid	Specifies the CHI NodeID for LDID # $\{i*4\}$	RO	11'h0

por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127

This register repeats 127 times. It parametrized by the i from 0 to 127. Specifies the mapping of RN-F's overridden LDID for default LDIDs # $\{i*4\}$ to # $\{i*4+3\}$. Valid only if POR_MXP_RNF_CLUSTER_EN_PARAM is 1

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1800 + # $\{[0, 1, 2, \dots, 126, 127]*8\}$
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.
Secure group override	por_ccg_ra_secure_register_groups_override.ldid_ctl

The following image shows the higher register bit assignments.

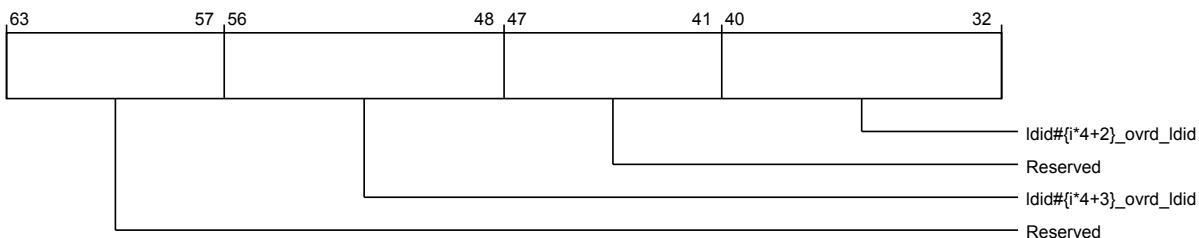


Figure 3-831 por_ccg_ra_por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (high)

The following table shows the por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 higher register bit assignments.

Table 3-851 por_ccg_ra_por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:48	ldid#{i*4+3}_ovrd_ldid	Specifies the Overridden LDID for Default LDID # $\{i*4+3\}$	RW	Configuration dependent
47:41	Reserved	Reserved	RO	-
40:32	ldid#{i*4+2}_ovrd_ldid	Specifies the Overridden LDID for Default LDID # $\{i*4+2\}$	RW	Configuration dependent

The following image shows the lower register bit assignments.

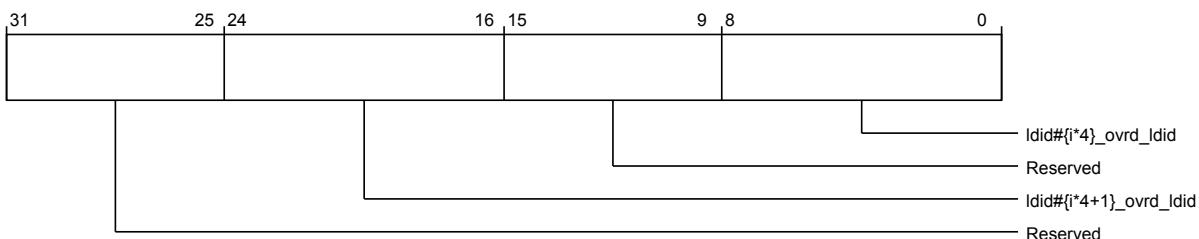


Figure 3-832 por_ccg_ra_por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (low)

The following table shows the por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 lower register bit assignments.

Table 3-852 por_ccg_ra_por_ccg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24:16	ldid#{i*4+1}_ovrd_ldid	Specifies the Overridden LDID for Default LDID #{i*4+1}	RW	Configuration dependent
15:9	Reserved	Reserved	RO	-
8:0	ldid#{i*4}_ovrd_ldid	Specifies the Overridden LDID for Default LDID #{i*4}	RW	Configuration dependent

por_ccg_ra_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

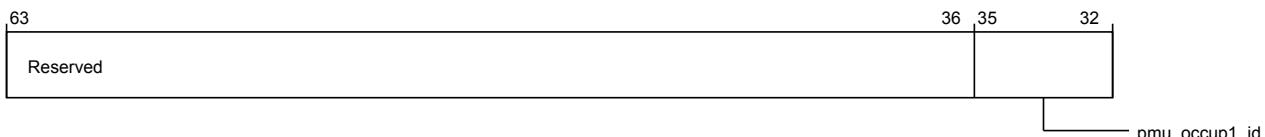


Figure 3-833 por_ccg_ra_por_ccg_ra_pmu_event_sel (high)

The following table shows the por_ccg_ra_pmu_event_sel higher register bit assignments.

Table 3-853 por_ccg_ra_por_ccg_ra_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:36	Reserved	Reserved	RO	-
35:32	pmu_occup1_id	PMU occupancy event selector ID	RW	4'b0

The following image shows the lower register bit assignments.

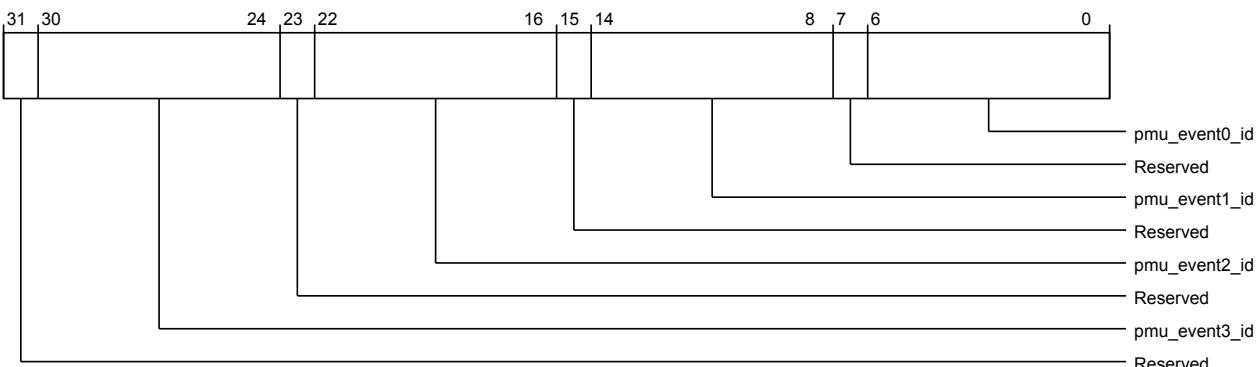


Figure 3-834 por_ccg_ra_por_ccg_ra_pmu_event_sel (low)

The following table shows the por ccg ra pmu event sel lower register bit assignments.

Table 3-854 por_ccg_ra_por_ccg_ra_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:24	pmu_event3_id	CXRA PMU Event 3 ID; see pmu_event0_id for encodings	RW	7'b0
23	Reserved	Reserved	RO	-
22:16	pmu_event2_id	CXRA PMU Event 2 ID; see pmu_event0_id for encodings	RW	7'b0
15	Reserved	Reserved	RO	-
14:8	pmu_event1_id	CXRA PMU Event 1 ID; see pmu_event0_id for encodings	RW	7'b0

Table 3-854 por_ccg_ra_por_ccg_ra_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7	Reserved	Reserved	RO	-
6:0	pmu_event0_id	CXRA PMU Event 0 ID 7'h00: No event 7'h41: Request Tracker (RHT) occupancy count overflow 7'h42: Snoop Tracker (SHT) occupancy count overflow 7'h43: Read Data Buffer (RDB) occupancy count overflow 7'h44: Write Data Buffer (WDB) occupancy count overflow 7'h45: Snoop Sink Buffer (SSB) occupancy count overflow 7'h46: CCIX RX broadcast snoops 7'h47: CCIX TX request chain 7'h48: CCIX TX request chain average length 7'h49: CHI internal RSP stall 7'h4A: CHI internal DAT stall 7'h4B: CCIX REQ Protocol credit Link 0 stall 7'h4C: CCIX REQ Protocol credit Link 1 stall 7'h4D: CCIX REQ Protocol credit Link 2 stall 7'h4E: CCIX DAT Protocol credit Link 0 stall 7'h4F: CCIX DAT Protocol credit Link 1 stall 7'h50: CCIX DAT Protocol credit Link 2 stall 7'h51: CHI external RSP stall 7'h52: CHI external DAT stall 7'h53: CCIX MISC Protocol credit Link 0 stall 7'h54: CCIX MISC Protocol credit Link 1 stall 7'h55: CCIX MISC Protocol credit Link 2 stall 7'h56: Request Tracker (RHT) allocations 7'h57: Snoop Tracker (SHT) allocations 7'h58: Read Data Buffer (RDB) allocations 7'h59: Write Data Buffer (WDB) allocations 7'h5A: Snoop Sink Buffer (SSB) allocations	RW	7'b0

por_ccg_ra_ccprtcl_link0_ctl

Functions as the CXRA CCIX Protocol Link 0 control register. Works with por_ccg_ra_exprtcl_link0_status.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset	16'h1C00
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

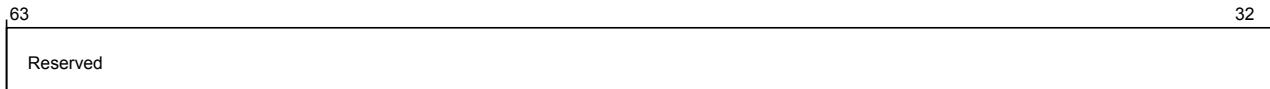


Figure 3-835 por_ccg_ra_por_ccg_ra_ccprtcl_link0_ctl (high)

The following table shows the por_ccg_ra_ccprtcl_link0_ctl higher register bit assignments.

Table 3-855 por_ccg_ra_por_ccg_ra_ccprtcl_link0_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

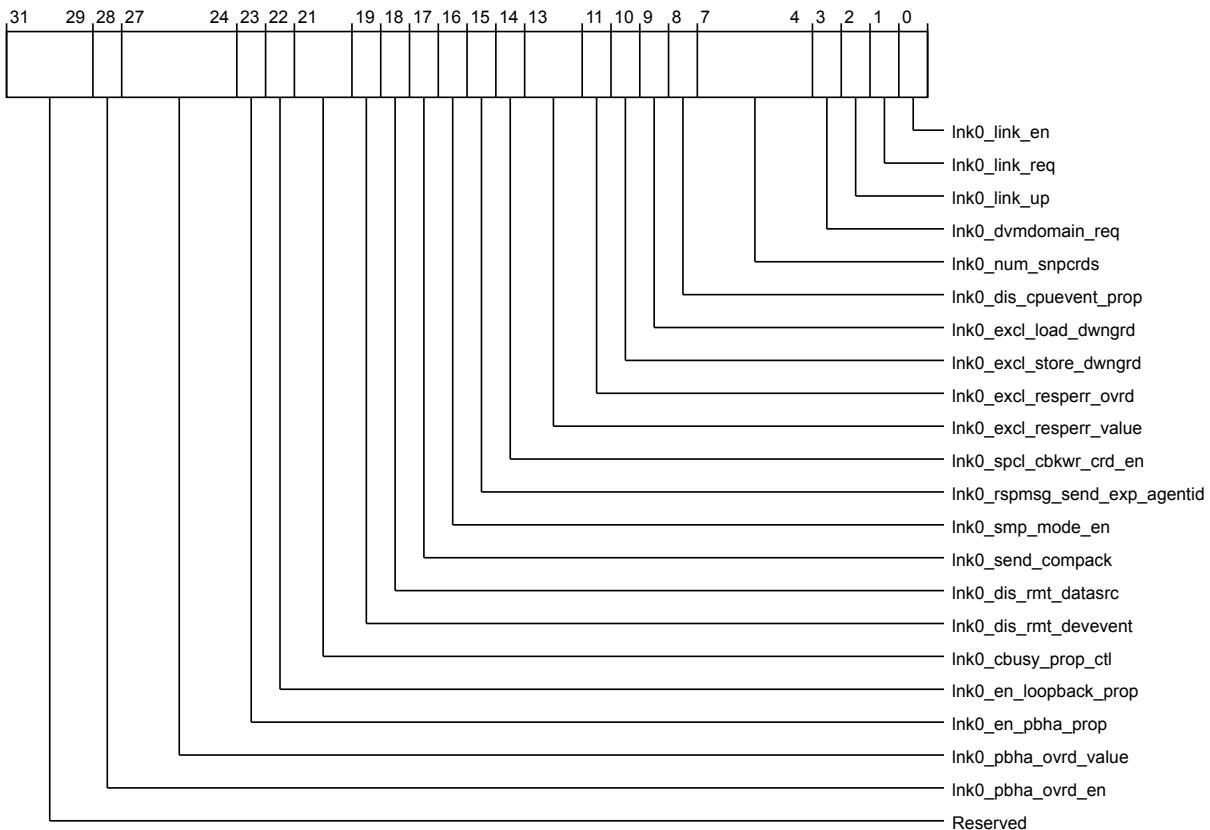


Figure 3-836 por_ccg_ra_por_ccg_ra_ccprtcl_link0_ctl (low)

The following table shows the por_ccg_ra_ccprtcl_link0_ctl lower register bit assignments.

Table 3-856 por_ccg_ra_por_ccg_ra_ccprtcl_link0_ctl (low)

Bits	Field name	Description	Type	Reset
31:29	Reserved	Reserved	RO	-
28	lnk0_pbha_ovrd_en	When set, overrides PBHA on CCIX Link 0. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	1'b0
27:24	lnk0_pbha_ovrd_value	Override value for PBHA on CCIX Link 0. Applicable only when lnk0_pbha_ovrd_en is set. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	4'b0
23	lnk0_en_pbha_prop	When set, enables propagation of PBHA on CCIX Link 0.	RW	1'b1
22	lnk0_en_loopback_prop	When set, enables propagation of LoopBack on CCIX Link 0.	RW	1'b1
21:20	lnk0_cbusy_prop_ctl	Controls the propagation of Cbusy field for CCIX Link 0. 2'b00: Send RA Cbusy on all responses based on the limits programmed in por_ccg_ra_cbusy_limit_ctl 2'b01: Pass through remote CBusy on late completion responses (CompData, Comp) 2'b10: Greater of RA Cbusy or remote Cbusy. Applicable to responses where remote Cbusy can be sent NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	2'b0
19	lnk0_dis_rmt_devevent	When set, disables propagation of remote Dev Event field for CCIX Link 0. NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
18	lnk0_dis_rmt_datasrc	When set, disables propagation of remote data source for CCIX Link 0. NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
17	lnk0_send_compack	When set, sends CompAck for CCIX Link 0.	RW	1'b0
16	lnk0_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 0.	RW	Configuration dependent
15	lnk0_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 0	RW	1'b0
14	lnk0_spcl_cbkwr_crd_en	When set, notifies RA to use special credits from HA to send CopyBack writes on CCIX Link 0 NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0

Table 3-856 por_ccg_ra_por_ccg_ra_ccprtcl_link0_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
13:12	lnk0_excl_resperr_value	<p>Two bit value to override RespErr field of an exclusive response. Applicable only if lnk0_excl_resperr_ovrd bit is set.</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	2'b0
11	lnk0_excl_resperr_ovrd	<p>When set, overrides the RespErr field of exclusive response with the lnk0_excl_resperr_value field</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
10	lnk0_excl_store_dwngrd	<p>When set, downgrades shareable exclusive store to shareable store when sending on CCIX Link 0</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
9	lnk0_excl_load_dwngrd	<p>When set, downgrades shareable exclusive load to shareable load when sending on CCIX Link 0</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
8	lnk0_dis_cpuevent_prop	<p>When set, disables the propagation of CPU Events on CCIX Link 0</p> <p>NOTE: This field is applicable only when SMP Mode enable is set.</p>	RW	1'b0
7:4	lnk0_num_snpercds	<p>Controls the number of CCIX snoop credits assigned to Link 0</p> <p>4'h0: Total credits are equally divided across all links</p> <p>4'h1: 25% of credits assigned</p> <p>4'h2: 50% of credits assigned</p> <p>4'h3: 75% of credits assigned</p> <p>4'h4: 100% of credits assigned</p> <p>4'hF: 0% of credits assigned</p>	RW	4'b0
3	lnk0_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for CCIX Link 0	RW	1'b0

Table 3-856 por_ccg_ra_por_ccg_ra_ccprtcl_link0_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	lnk0_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk0_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk0_link_en	<p>Enables CCIX Link 0 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_ccg_ra_cxprtcl_link0_status

Functions as the CXRA CCIX Protocol Link 0 status register. Works with por_ccg_ra_ccprtcl_link0_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C08

Register reset 64'b0010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-837 por_ccg_ra_por_ccg_ra_cxprtcl_link0_status (high)

The following table shows the por_ccg_ra_cxprtcl_link0_status higher register bit assignments.

Table 3-857 por_ccg_ra_por_ccg_ra_cxprtcl_link0_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

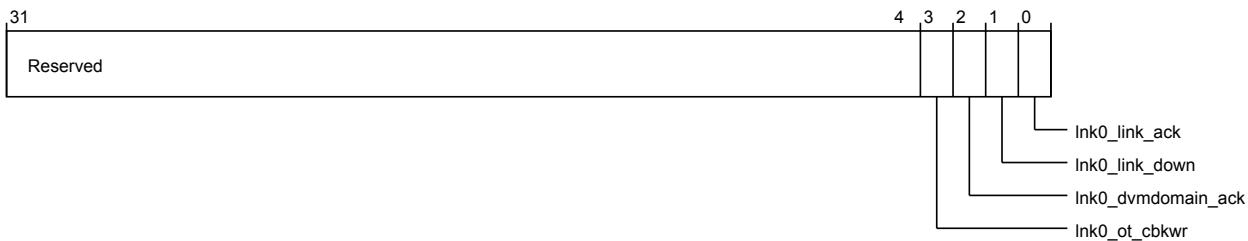


Figure 3-838 por_ccg_ra_por_ccg_ra_cxprtcl_link0_status (low)

The following table shows the por_ccg_ra_por_ccg_ra_cxprtcl_link0_status lower register bit assignments.

Table 3-858 por_ccg_ra_por_ccg_ra_cxprtcl_link0_status (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	lnk0_ot_cbkwr	Provides status for outstanding CopyBack Write for CCIX Link0	RO	1'b0
2	lnk0_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for CCIX Link 0	RO	1'b0
1	lnk0_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	lnk0_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops granting protocol credits and starts returning protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_ccg_ra_ccprtcl_link1_ctl

Functions as the CXRA CCIX Protocol Link 1 control register. Works with por_ccg_ra_cxprtcl_link1_status.

Its characteristics are:

Type	RW
Register width (Bits)	64

Address offset	16'h1C10
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

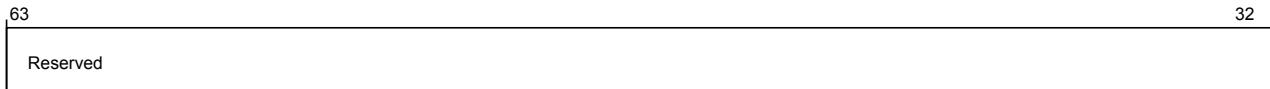


Figure 3-839 por_ccg_ra_por_ccg_ra_ccprtcl_link1_ctl (high)

The following table shows the por_ccg_ra_ccprtcl_link1_ctl higher register bit assignments.

Table 3-859 por_ccg_ra_por_ccg_ra_ccprtcl_link1_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

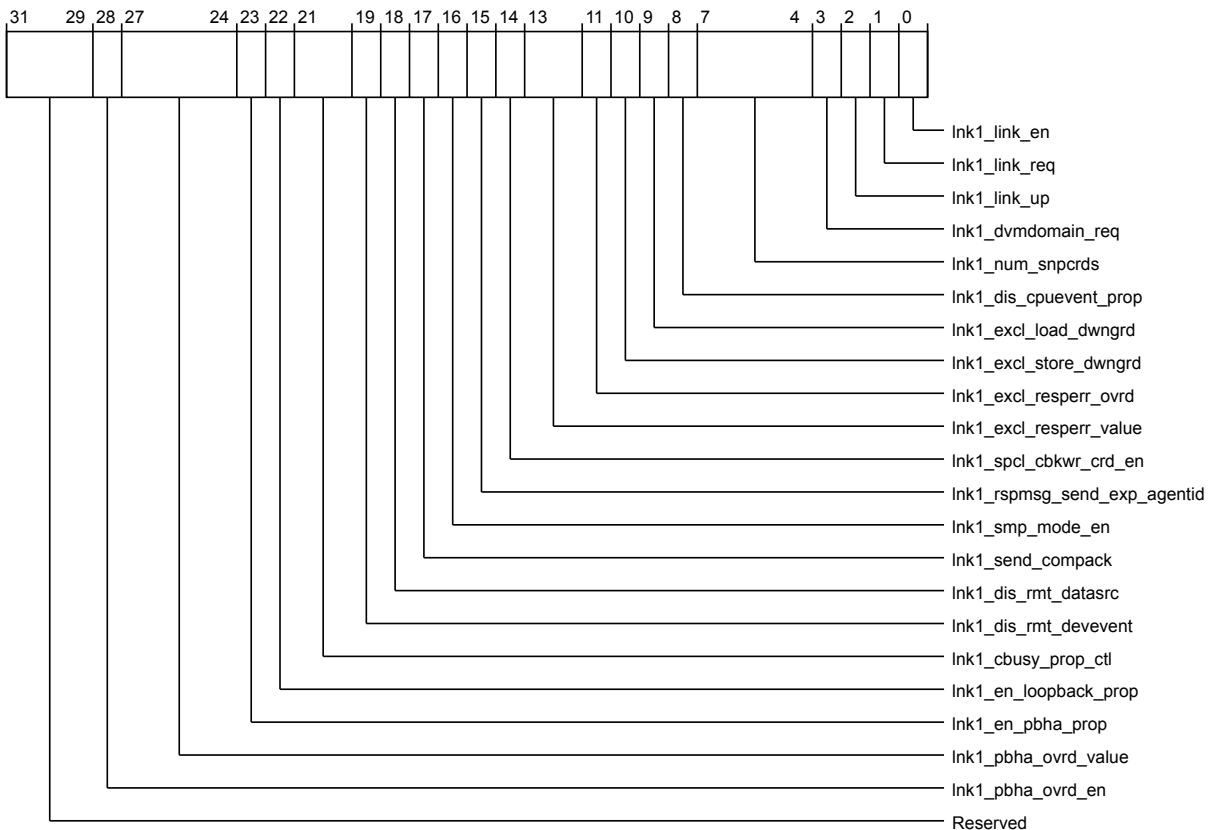


Figure 3-840 por_ccg_ra_por_ccg_ra_ccprtcl_link1_ctl (low)

The following table shows the por_ccg_ra_ccprtcl_link1_ctl lower register bit assignments.

Table 3-860 por_ccg_ra_por_ccg_ra_ccprtcl_link1_ctl (low)

Bits	Field name	Description	Type	Reset
31:29	Reserved	Reserved	RO	-
28	lnk1_pbha_ovrd_en	When set, overrides PBHA on CCIX Link 1. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	1'b0
27:24	lnk1_pbha_ovrd_value	Override value for PBHA on CCIX Link 1. Applicable only when lnk1_pbha_ovrd_en is set. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	4'b0
23	lnk1_en_pbha_prop	When set, enables propagation of PBHA on CCIX Link 1.	RW	1'b1
22	lnk1_en_loopback_prop	When set, enables propagation of LoopBack on CCIX Link 1.	RW	1'b1
21:20	lnk1_cbusy_prop_ctl	Controls the propagation of Cbusy field for CCIX Link 1. 2'b00: Send RA Cbusy on all responses based on the limits programmed in por_ccg_ra_cbusy_limit_ctl 2'b01: Pass through remote CBusy on late completion responses (CompData, Comp) 2'b10: Greater of RA Cbusy or remote Cbusy. Applicable to responses where remote Cbusy can be sent NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	2'b0
19	lnk1_dis_rmt_devevent	When set, disables propagation of remote Dev Event field for CCIX Link 1. NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
18	lnk1_dis_rmt_datasrc	When set, disables propagation of remote data source for CCIX Link 1. NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
17	lnk1_send_compack	When set, sends CompAck for CCIX Link 1.	RW	1'b0
16	lnk1_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 1.	RW	Configuration dependent
15	lnk1_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 1	RW	1'b0
14	lnk1_spcl_cbkwr_crd_en	When set, notifies RA to use special credits from HA to send CopyBack writes on CCIX Link 1 NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0

Table 3-860 por_ccg_ra_por_ccg_ra_ccprtcl_link1_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
13:12	lnk1_excl_resperr_value	<p>Two bit value to override RespErr field of an exclusive response. Applicable only if lnk1_excl_resperr_ovrd bit is set.</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	2'b0
11	lnk1_excl_resperr_ovrd	<p>When set, overrides the RespErr field of exclusive response with the lnk1_excl_resperr_value field</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
10	lnk1_excl_store_dwngrd	<p>When set, downgrades shareable exclusive store to shareable store when sending on CCIX Link 1</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
9	lnk1_excl_load_dwngrd	<p>When set, downgrades shareable exclusive load to shareable load when sending on CCIX Link 1</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
8	lnk1_dis_cpuevent_prop	<p>When set, disables the propagation of CPU Events on CCIX Link 1</p> <p>NOTE: This field is applicable only when SMP Mode enable parameter is set.</p>	RW	1'b0
7:4	lnk1_num_snpcrds	<p>Controls the number of CCIX snoop credits assigned to Link 1</p> <p>4'h0: Total credits are equally divided across all links</p> <p>4'h1: 25% of credits assigned</p> <p>4'h2: 50% of credits assigned</p> <p>4'h3: 75% of credits assigned</p> <p>4'h4: 100% of credits assigned</p> <p>4'hF: 0% of credits assigned</p>	RW	4'b0
3	lnk1_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for CCIX Link 1	RW	1'b0

Table 3-860 por_ccg_ra_por_ccg_ra_ccprtcl_link1_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	lnk1_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk1_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk1_link_en	<p>Enables CCIX Link 1 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_ccg_ra_cxprtcl_link1_status

Functions as the CXRA CCIX Protocol Link 1 status register. Works with por_ccg_ra_ccprtcl_link1_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C18

Register reset 64'b0010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-841 por_ccg_ra_por_ccg_ra_cxprtcl_link1_status (high)

The following table shows the por_ccg_ra_cxprtcl_link1_status higher register bit assignments.

Table 3-861 por_ccg_ra_por_ccg_ra_cxprtcl_link1_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

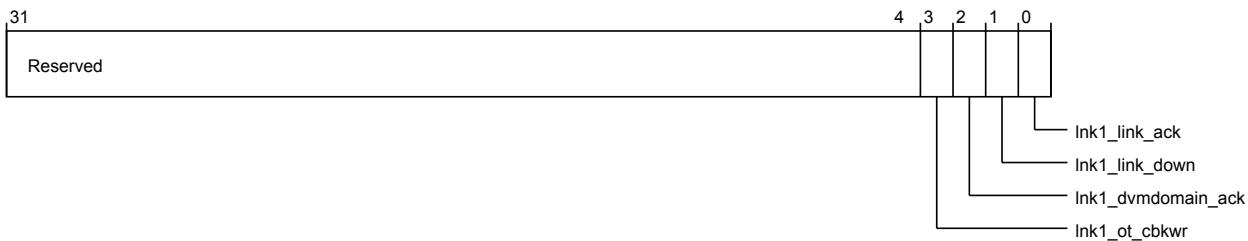


Figure 3-842 por_ccg_ra_por_ccg_ra_cxprtcl_link1_status (low)

The following table shows the por_ccg_ra_por_ccg_ra_cxprtcl_link1_status lower register bit assignments.

Table 3-862 por_ccg_ra_por_ccg_ra_cxprtcl_link1_status (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	lnk1_ot_cbkwr	Provides status for outstanding CopyBack Write for CCIX Link1	RO	1'b0
2	lnk1_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for CCIX Link 1	RO	1'b0
1	lnk1_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	lnk1_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

por_ccg_ra_ccprtcl_link2_ctl

Functions as the CXRA CCIX Protocol Link 2 control register. Works with por_ccg_ra_cxprtcl_link2_status.

Its characteristics are:

Type	RW
Register width (Bits)	64

Address offset	16'h1C20
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.

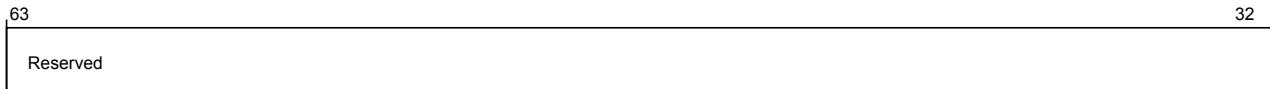


Figure 3-843 por_ccg_ra_por_ccg_ra_ccprtcl_link2_ctl (high)

The following table shows the por_ccg_ra_ccprtcl_link2_ctl higher register bit assignments.

Table 3-863 por_ccg_ra_por_ccg_ra_ccprtcl_link2_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

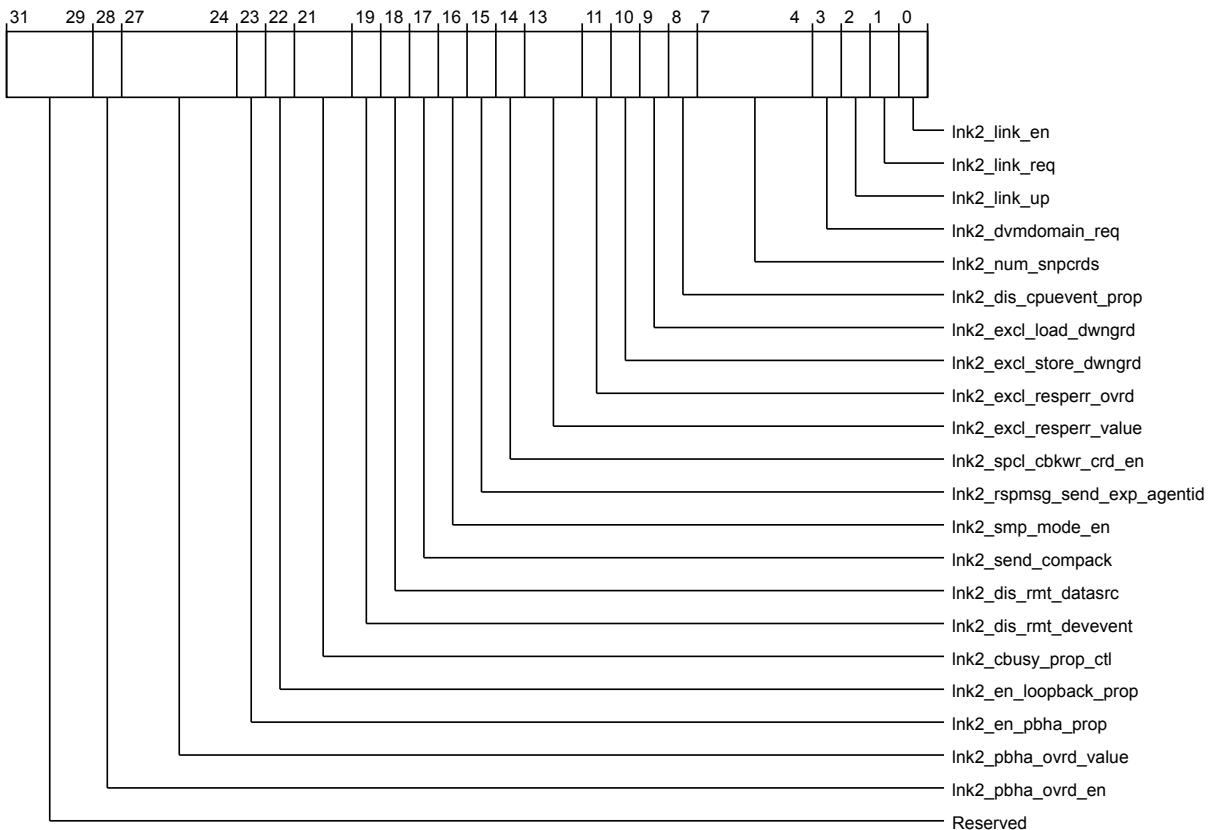


Figure 3-844 por_ccg_ra_por_ccg_ra_ccprtcl_link2_ctl (low)

The following table shows the por_ccg_ra_ccprtcl_link2_ctl lower register bit assignments.

Table 3-864 por_ccg_ra_por_ccg_ra_ccprtcl_link2_ctl (low)

Bits	Field name	Description	Type	Reset
31:29	Reserved	Reserved	RO	-
28	lnk2_pbha_ovrd_en	When set, overrides PBHA on CCIX Link 2. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	1'b0
27:24	lnk2_pbha_ovrd_value	Override value for PBHA on CCIX Link 2. Applicable only when lnk2_pbha_ovrd_en is set. Note: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode)	RW	4'b0
23	lnk2_en_pbha_prop	When set, enables propagation of PBHA on CCIX Link 2.	RW	1'b1
22	lnk2_en_loopback_prop	When set, enables propagation of LoopBack on CCIX Link 2.	RW	1'b1
21:20	lnk2_cbusy_prop_ctl	Controls the propagation of Cbusy field for CCIX Link 2. 2'b00: Send RA Cbusy on all responses based on the limits programmed in por_ccg_ra_cbusy_limit_ctl 2'b01: Pass through remote CBusy on late completion responses (CompData, Comp) 2'b10: Greater of RA Cbusy or remote Cbusy. Applicable to responses where remote Cbusy can be sent NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	2'b0
19	lnk2_dis_rmt_devevent	When set, disables propagation of remote Dev Event field for CCIX Link 2. NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
18	lnk2_dis_rmt_datasrc	When set, disables propagation of remote data source for CCIX Link 2. NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0
17	lnk2_send_compack	When set, sends CompAck for CCIX Link 2.	RW	1'b0
16	lnk2_smp_mode_en	When set, enables Symmetric Multiprocessor Mode (SMP) Mode for CCIX Link 2.	RW	Configuration dependent
15	lnk2_rspmsg_send_exp_agentid	When set sends Expanded Agent ID on CCIX Response Messages for CCIX Link 2	RW	1'b0
14	lnk2_spcl_cbkwr_crd_en	When set, notifies RA to use special credits from HA to send CopyBack writes on CCIX Link 2 NOTE: This field is applicable only if the link programmed for SMP mode (i.e. SMP Mode enable bit is set)	RW	1'b0

Table 3-864 por_ccg_ra_por_ccg_ra_ccprtcl_link2_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
13:12	lnk2_excl_resperr_value	<p>Two bit value to override RespErr field of an exclusive response. Applicable only if lnk2_excl_resperr_ovrd bit is set.</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	2'b0
11	lnk2_excl_resperr_ovrd	<p>When set, overrides the RespErr field of exclusive response with the lnk2_excl_resperr_value field</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
10	lnk2_excl_store_dwngrd	<p>When set, downgrades shareable exclusive store to shareable store when sending on CCIX Link 2</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
9	lnk2_excl_load_dwngrd	<p>When set, downgrades shareable exclusive load to shareable load when sending on CCIX Link 2</p> <p>NOTE: This field is applicable only when SMP Mode enable bit is clear (i.e. Non-SMP mode) and must only be used if the corresponding link end pair does not support exclusive monitoring</p>	RW	1'b0
8	lnk2_dis_cpuevent_prop	<p>When set, disables the propagation of CPU Events on CCIX Link 2</p> <p>NOTE: This field is applicable only when SMP Mode enable parameter is set.</p>	RW	1'b0
7:4	lnk2_num_snpcrds	<p>Controls the number of CCIX snoop credits assigned to Link 2</p> <p>4'h0: Total credits are equally divided across all links</p> <p>4'h1: 25% of credits assigned</p> <p>4'h2: 50% of credits assigned</p> <p>4'h3: 75% of credits assigned</p> <p>4'h4: 100% of credits assigned</p> <p>4'hF: 0% of credits assigned</p>	RW	4'b0
3	lnk2_dvmdomain_req	Controls DVM domain enable (SYSCOREQ) for CCIX Link 2	RW	1'b0

Table 3-864 por_ccg_ra_por_ccg_ra_ccprtcl_link2_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	lnk2_link_up	<p>Link Up status. Software writes this register bit to indicate Link status after polling Link_ACK and Link_DN status in the remote agent</p> <p>1'b0: Link is not Up. Software clears Link_UP when Link_ACK status is clear and Link_DN status is set in both local and remote agents. The local agent stops responding to any protocol activity from remote agent, including acceptance of protocol credits, when Link_UP is clear</p> <p>1'b1: Link is Up. Software sets Link_UP when Link_ACK status is set and Link_DN status is clear in both local and remote agents; the local agent starts sending local protocol credits to remote agent</p>	RW	1'b0
1	lnk2_link_req	<p>Link Up/Down request; software writes this register bit to request a Link Up or Link Down in the local agent</p> <p>1'b0: Link Down request</p> <p>NOTE: The local agent does not return remote protocol credits yet since remote agent may still be in Link_UP state.</p> <p>1'b1: Link Up request</p>	RW	1'b0
0	lnk2_link_en	<p>Enables CCIX Link 2 when set</p> <p>1'b0: Link is disabled</p> <p>1'b1: Link is enabled</p>	RW	1'b0

por_ccg_ra_cxprtcl_link2_status

Functions as the CXRA CCIX Protocol Link 2 status register. Works with por_ccg_ra_ccprtcl_link2_ctl.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1C28

Register reset 64'b0010

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-845 por_ccg_ra_por_ccg_ra_cxprtcl_link2_status (high)

The following table shows the por_ccg_ra_cxprtcl_link2_status higher register bit assignments.

Table 3-865 por_ccg_ra_por_ccg_ra_cxprtcl_link2_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

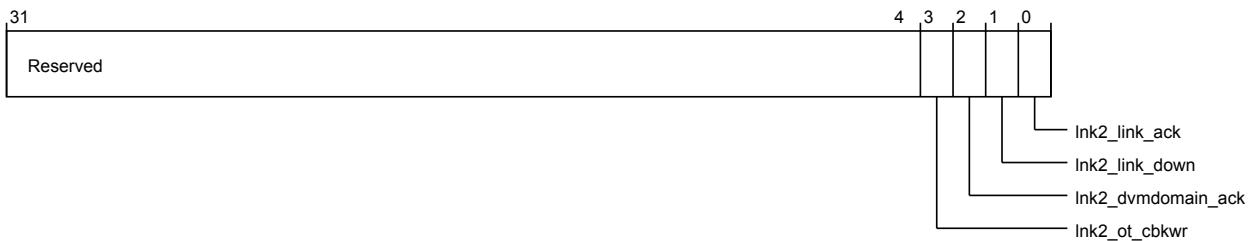


Figure 3-846 por_ccg_ra_por_ccg_ra_cxprtcl_link2_status (low)

The following table shows the por_ccg_ra_por_ccg_ra_cxprtcl_link2_status lower register bit assignments.

Table 3-866 por_ccg_ra_por_ccg_ra_cxprtcl_link2_status (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	lnk2_ot_cbkwr	Provides status for outstanding CopyBack Write for CCIX Link2	RO	1'b0
2	lnk2_dvmdomain_ack	Provides DVM domain status (SYSCOACK) for CCIX Link 2	RO	1'b0
1	lnk2_link_down	Link Down status; hardware updates this register bit to indicate Link Down status 1'b0: Link is not Down; hardware clears Link_DN when it receives a Link Up request 1'b1: Link is Down; hardware sets Link_DN after the local agent has received all local protocol credits. The local agent must continue to respond to any remote protocol activity, including accepting and returning remote protocol credits until Link Up is clear	RO	1'b1
0	lnk2_link_ack	Link Up/Down acknowledge; hardware updates this register bit to acknowledge the software link request 1'b0: Link Down acknowledge; hardware clears Link_ACK on receiving a Link Down request; the local agent stops sending protocol credits to the remote agent when Link_ACK is clear 1'b1: Link Up acknowledge; hardware sets Link_ACK when the local agent is ready to start accepting protocol credits from the remote agent NOTE: The local agent must clear Link_DN before setting Link_ACK.	RO	1'b0

3.3.14 Debug and trace register descriptions

This section lists the debug and trace registers.

por_dt_node_info

Provides component identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h0
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

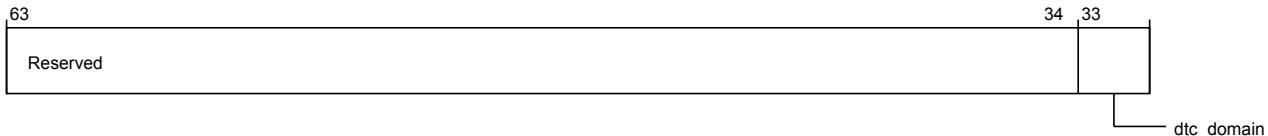


Figure 3-847 por_dt_por_dt_node_info (high)

The following table shows the por_dt_node_info higher register bit assignments.

Table 3-867 por_dt_por_dt_node_info (high)

Bits	Field name	Description	Type	Reset
63:34	Reserved	Reserved	RO	-
33:32	dtc_domain	DTC domain number	RO	Configuration dependent

The following image shows the lower register bit assignments.

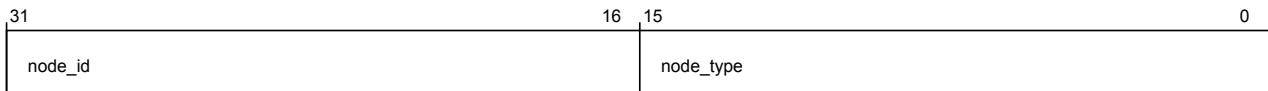


Figure 3-848 por_dt_por_dt_node_info (low)

The following table shows the por_dt_node_info lower register bit assignments.

Table 3-868 por_dt_por_dt_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component CHI node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h3

por_dt_child_info

Provides component child identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

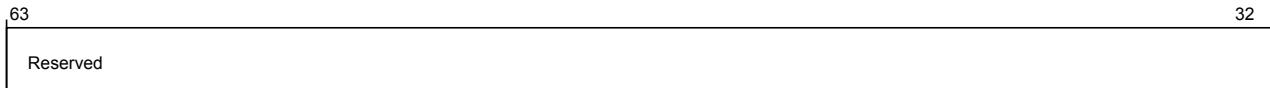


Figure 3-849 por_dt_por_dt_child_info (high)

The following table shows the por_dt_child_info higher register bit assignments.

Table 3-869 por_dt_por_dt_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

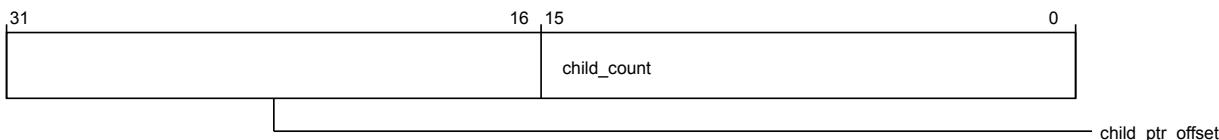


Figure 3-850 por_dt_por_dt_child_info (low)

The following table shows the por_dt_child_info lower register bit assignments.

Table 3-870 por_dt_por_dt_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0000
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_dt_secure_access

Functions as the secure access control register.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

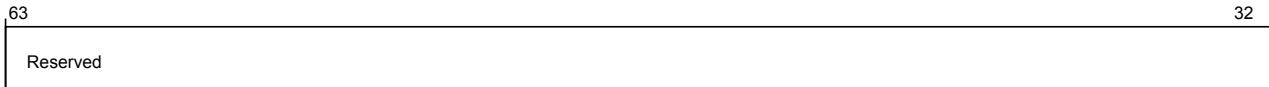


Figure 3-851 por_dt_por_dt_secure_access (high)

The following table shows the por_dt_secure_access higher register bit assignments.

Table 3-871 por_dt_por_dt_secure_access (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

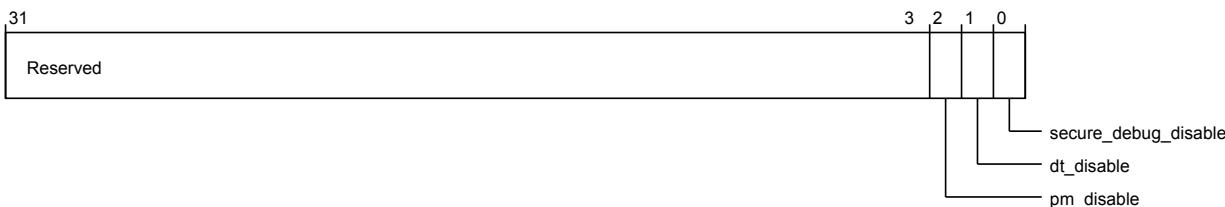


Figure 3-852 por_dt_por_dt_secure_access (low)

The following table shows the por_dt_secure_access lower register bit assignments.

Table 3-872 por_dt_por_dt_secure_access (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	pm_disable	PMU disable 1'b0: PMU function is not affected 1'b1: PMU function is disabled.	RW	1'b0
1	dt_disable	Debug disable 1'b0: DT function is not affected 1'b1: DT function is disabled.	RW	1'b0
0	secure_debug_disable	Secure debug disable 1'b0: Secure events are monitored by the PMU 1'b1: Secure events are only monitored by the PMU if SPNIDEN is set to 1	RW	1'b0

por_dt_dtc_ctl

Functions as the debug trace control register.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA00

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-853 por_dt_por_dt_dtc_ctl (high)

The following table shows the por_dt_dtc_ctl higher register bit assignments.

Table 3-873 por_dt_por_dt_dtc_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

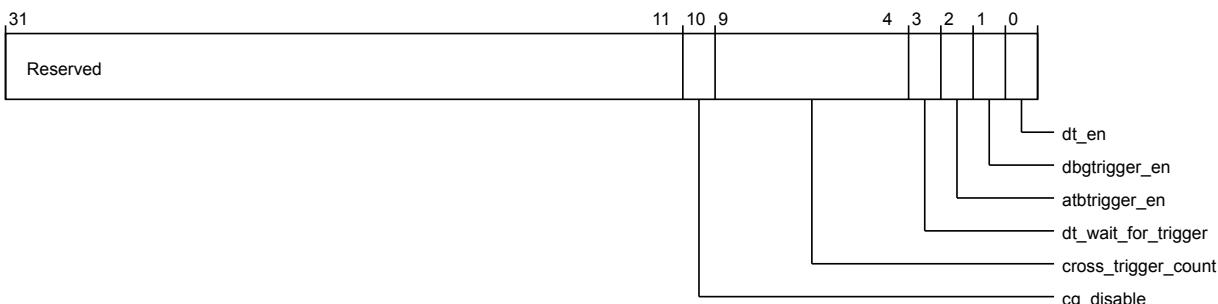


Figure 3-854 por dt por dt dtc ctl (low)

The following table shows the por dt dtc ctl lower register bit assignments.

Table 3-874 por dt por dt dtc ctl (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10	cg_disable	Disables DT architectural clock gates	RW	1'b0
9:4	cross_trigger_count	Number of cross triggers received before trace enable NOTE: Only applicable if dt_wait_for_trigger is set to 1.	RW	6'b0
3	dt_wait_for_trigger	Enables waiting for cross trigger before trace enable	RW	1'b0
2	atbtrigger_en	ATB trigger enable	RW	1'b0

Table 3-874 por_dt_por_dt_dtc_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
1	dbgtrigger_en	DBGWATCHTRIG enable	RW	1'b0
0	dt_en	Enables debug, trace, and PMU features	RW	1'b0

por_dt_trigger_status

Provides the trigger status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hA10

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-855 por_dt_por_dt_trigger_status (high)

The following table shows the por_dt_trigger_status higher register bit assignments.

Table 3-875 por_dt_por_dt_trigger_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

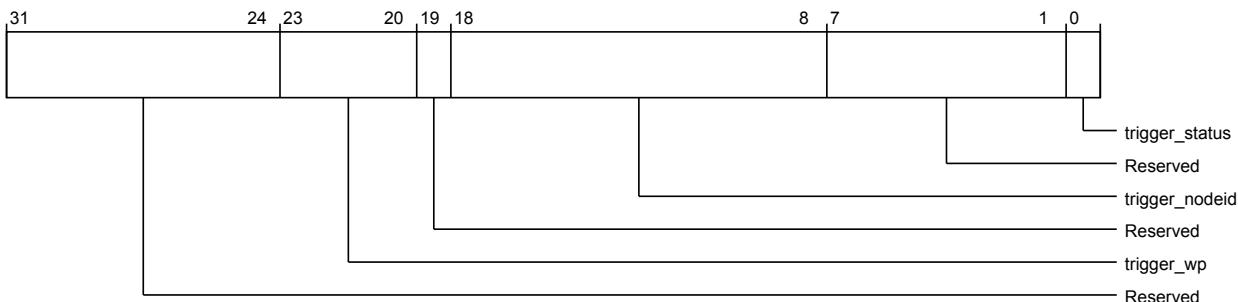


Figure 3-856 por_dt_por_dt_trigger_status (low)

The following table shows the por_dt_trigger_status lower register bit assignments.

Table 3-876 por_dt_por_dt_trigger_status (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:20	trigger_wp	DBGWATCHTRIGREQ assertion and/or ATB trigger are caused by watchpoint	RO	1'h0
19	Reserved	Reserved	RO	-
18:8	trigger_nodeid	DBGWATCHTRIGREQ assertion and/or ATB trigger are caused by node ID	RO	11'h0
7:1	Reserved	Reserved	RO	-
0	trigger_status	Indicates DBGWATCHTRIGREQ assertion and/or ATB trigger	RO	1'h0

por_dt_trigger_status_clr

Clears the trigger status.

Its characteristics are:

Type WO

Register width (Bits) 64

Address offset 16'hA20

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

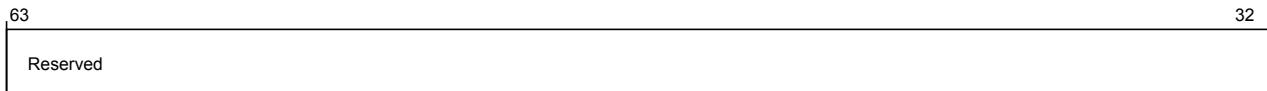


Figure 3-857 por_dt_por_dt_trigger_status_clr (high)

The following table shows the por_dt_trigger_status_clr higher register bit assignments.

Table 3-877 por_dt_por_dt_trigger_status_clr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

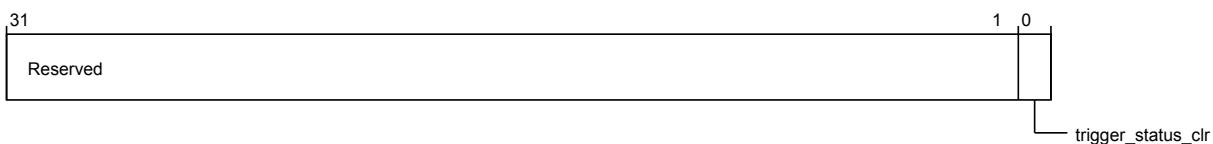


Figure 3-858 por_dt_por_dt_trigger_status_clr (low)

The following table shows the por_dt_trigger_status_clr lower register bit assignments.

Table 3-878 por_dt_por_dt_trigger_status_clr (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	trigger_status_clr	Write a 1 to clear por_dt_trigger_status.trigger_status	WO	1'b0

por_dt_trace_control

Functions as the trace control register.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA30

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

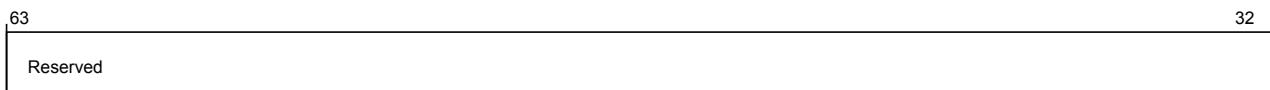


Figure 3-859 por_dt_por_dt_trace_control (high)

The following table shows the por_dt_trace_control higher register bit assignments.

Table 3-879 por_dt_por_dt_trace_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

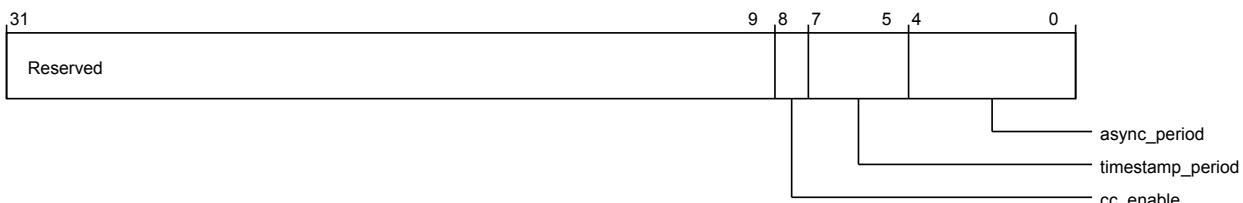


Figure 3-860 por_dt_por_dt_trace_control (low)

The following table shows the por_dt_trace_control lower register bit assignments.

Table 3-880 por_dt_por_dt_trace_control (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	cc_enable	Cycle count enable	RW	1'b0
7:5	timestamp_period	Time stamp packet insertion period 3'b000: Time stamp disabled 3'b011: Time stamp every 8K clock cycles 3'b100: Time stamp every 16K clock cycles 3'b101: Time stamp every 32K clock cycles 3'b110: Time stamp every 64K clock cycles	RW	3'b0
4:0	async_period	Alignment sync packet insertion period 5'h00: Alignment sync disabled 5'h08: Alignment sync inserted after 256B of trace 5'h09: Alignment sync inserted after 512B of trace 5'h14: Alignment sync inserted after 1048576B of trace NOTE: All other values are reserved.	RW	5'b0

por_dt_traceid

Contains the ATB ID.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA48

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-861 por_dt_por_dt_traceid (high)

The following table shows the por_dt_traceid higher register bit assignments.

Table 3-881 por_dt_por_dt_traceid (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

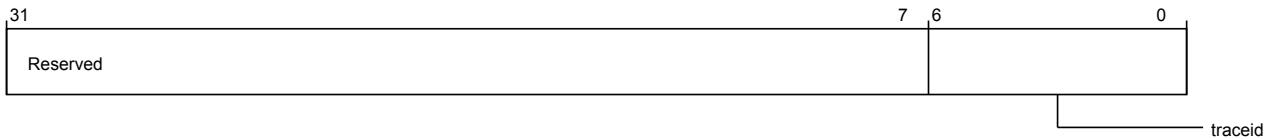


Figure 3-862 por_dt_por_dt_traceid (low)

The following table shows the por_dt_traceid lower register bit assignments.

Table 3-882 por_dt_por_dt_traceid (low)

Bits	Field name	Description	Type	Reset
31:7	Reserved	Reserved	RO	-
6:0	traceid	ATB ID	RW	7'h0

por_dt_pmevcntAB

Contains the PMU event counters A and B.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2000
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

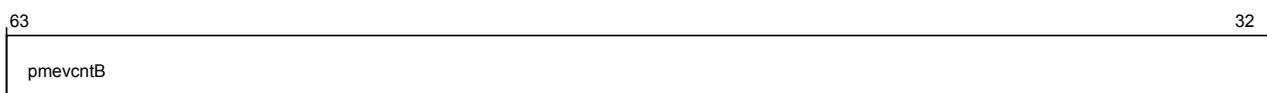


Figure 3-863 por_dt_por_dt_pmevcntab (high)

The following table shows the por_dt_pmevcntAB higher register bit assignments.

Table 3-883 por_dt_por_dt_pmevcntab (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntB	PMU counter B	RW	32'h0000

The following image shows the lower register bit assignments.



Figure 3-864 por_dt_por_dt_pmevcntab (low)

The following table shows the por_dt_pmevcntAB lower register bit assignments.

Table 3-884 por_dt_por_dt_pmevcntab (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntA	PMU counter A	RW	32'h0000

por_dt_pmevcntCD

Contains the PMU event counters C and D.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2010

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-865 por_dt_por_dt_pmevcntcd (high)

The following table shows the por_dt_pmevcntCD higher register bit assignments.

Table 3-885 por_dt_por_dt_pmevcntcd (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntD	PMU counter D	RW	32'h0000

The following image shows the lower register bit assignments.

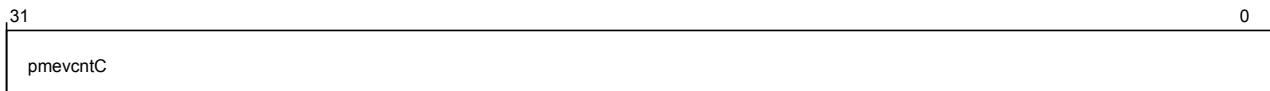


Figure 3-866 por_dt_por_dt_pmevcntcd (low)

The following table shows the por_dt_pmevcntCD lower register bit assignments.

Table 3-886 por_dt_por_dt_pmevcntcd (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntC	PMU counter C	RW	32'h0000

por_dt_pmevcntEF

Contains the PMU event counters E and F.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2020
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-867 por_dt_por_dt_pmevcntef (high)

The following table shows the por_dt_pmevcntEF higher register bit assignments.

Table 3-887 por_dt_por_dt_pmevcntef (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntF	PMU counter F	RW	32'h0000

The following image shows the lower register bit assignments.

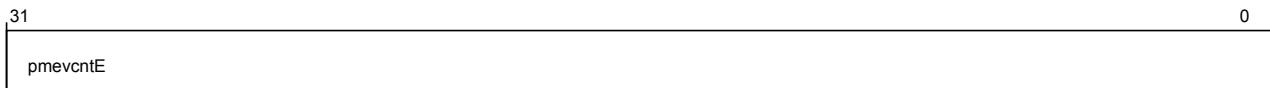


Figure 3-868 por_dt_por_dt_pmevcntef (low)

The following table shows the por_dt_pmevcntEF lower register bit assignments.

Table 3-888 por_dt_por_dt_pmevcntef (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntE	PMU counter E	RW	32'h0000

por_dt_pmevcntGH

Contains the PMU event counters G and H.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2030
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

63
pmevcntH

32

Figure 3-869 por_dt_por_dt_pmevcntgh (high)

The following table shows the por_dt_pmevcntGH higher register bit assignments.

Table 3-889 por_dt_por_dt_pmevcntgh (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntH	PMU counter H	RW	32'h0000

The following image shows the lower register bit assignments.

31
pmevcntG

0

Figure 3-870 por_dt_por_dt_pmevcntgh (low)

The following table shows the por_dt_pmevcntGH lower register bit assignments.

Table 3-890 por_dt_por_dt_pmevcntgh (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntG	PMU counter G	RW	32'h0000

por_dt_pmccntr

Contains the PMU cycle counter.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2040

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63
Reserved

40 39

32

pmccntr

Figure 3-871 por_dt_por_dt_pmccntr (high)

The following table shows the por_dt_pmccntr higher register bit assignments.

Table 3-891 por_dt_por_dt_pmccntr (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pmccntr	PMU cycle counter	RW	40'h0

The following image shows the lower register bit assignments.



Figure 3-872 por_dt_por_dt_pmccntr (low)

The following table shows the por_dt_pmccntr lower register bit assignments.

Table 3-892 por_dt_por_dt_pmccntr (low)

Bits	Field name	Description	Type	Reset
31:0	pmccntr	PMU cycle counter	RW	40'h0

por_dt_pmevcntsrAB

Contains the PMU event counter shadow registers A and B.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2050

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

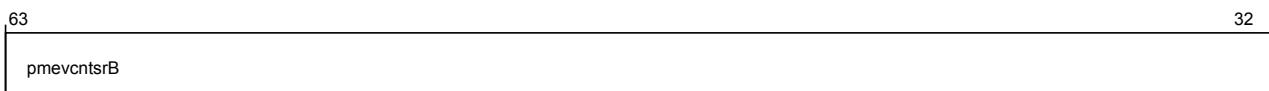


Figure 3-873 por_dt_por_dt_pmevcntsrab (high)

The following table shows the por_dt_pmevcntsrAB higher register bit assignments.

Table 3-893 por_dt_por_dt_pmevcntsrab (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntsrB	PMU counter B shadow register	RW	32'h0000

The following image shows the lower register bit assignments.



Figure 3-874 por_dt_por_dt_pmevcntsrab (low)

The following table shows the por_dt_pmevcntsrAB lower register bit assignments.

Table 3-894 por_dt_por_dt_pmevcntsrab (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntsrA	PMU counter A shadow register	RW	32'h0000

por_dt_pmevcntsrCD

Contains the PMU event counter shadow registers C and D.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2060

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

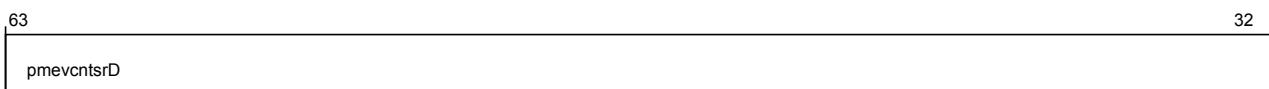


Figure 3-875 por_dt_por_dt_pmevcntsrcd (high)

The following table shows the por_dt_pmevcntsrCD higher register bit assignments.

Table 3-895 por_dt_por_dt_pmevcntsrcd (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntsrD	PMU counter D shadow register	RW	32'h0000

The following image shows the lower register bit assignments.

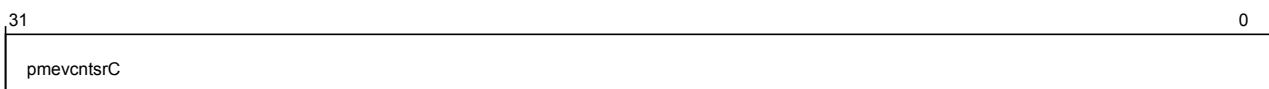


Figure 3-876 por_dt_por_dt_pmevcntsrcd (low)

The following table shows the por_dt_pmevcntsrCD lower register bit assignments.

Table 3-896 por_dt_por_dt_pmevcntsrefc (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntsrefC	PMU counter C shadow register	RW	32'h0000

por_dt_pmevcntsrefEF

Contains the PMU event counter shadow registers E and F.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2070

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

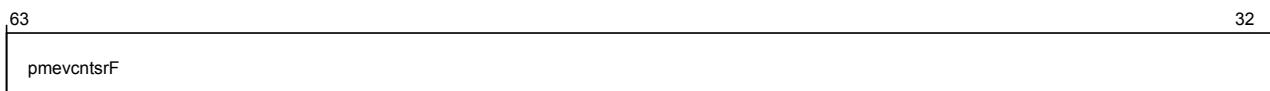


Figure 3-877 por_dt_por_dt_pmevcntsref (high)

The following table shows the por_dt_pmevcntsrefEF higher register bit assignments.

Table 3-897 por_dt_por_dt_pmevcntsref (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntsrefF	PMU counter F shadow register	RW	32'h0000

The following image shows the lower register bit assignments.

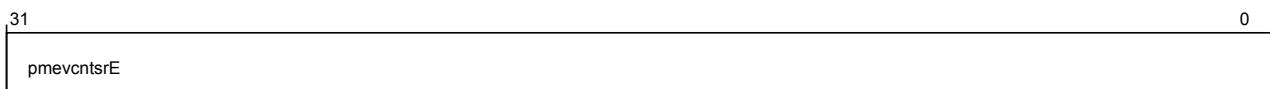


Figure 3-878 por_dt_por_dt_pmevcntsref (low)

The following table shows the por_dt_pmevcntsrefEF lower register bit assignments.

Table 3-898 por_dt_por_dt_pmevcntsref (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntsrefE	PMU counter E shadow register	RW	32'h0000

por_dt_pmevcntsrefGH

Contains the PMU event counter shadow registers G and H.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2080
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

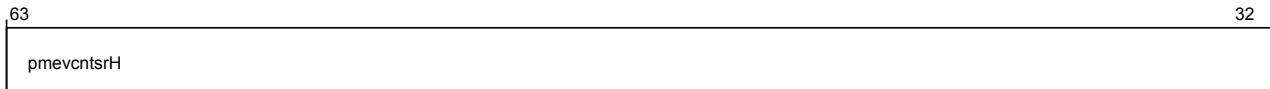


Figure 3-879 por_dt_por_dt_pmevcntsrgH (high)

The following table shows the por_dt_pmevcntsrgH higher register bit assignments.

Table 3-899 por_dt_por_dt_pmevcntsrgH (high)

Bits	Field name	Description	Type	Reset
63:32	pmevcntsrH	PMU counter H shadow register	RW	32'h0000

The following image shows the lower register bit assignments.

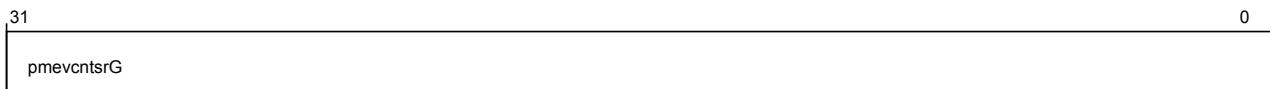


Figure 3-880 por_dt_por_dt_pmevcntsrgH (low)

The following table shows the por_dt_pmevcntsrgH lower register bit assignments.

Table 3-900 por_dt_por_dt_pmevcntsrgH (low)

Bits	Field name	Description	Type	Reset
31:0	pmevcntsrG	PMU counter G shadow register	RW	32'h0000

por_dt_pmccntrs

Contains the PMU cycle counter shadow register.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2090
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

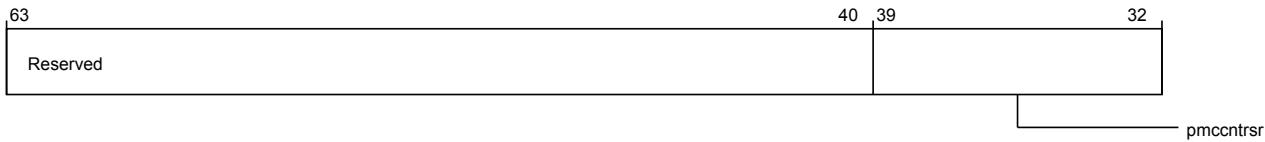


Figure 3-881 por_dt_por_dt_pmccntrs (high)

The following table shows the por_dt_pmccntrs higher register bit assignments.

Table 3-901 por_dt_por_dt_pmccntrs (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pmccntrs	PMU cycle counter shadow register	RW	40'h0

The following image shows the lower register bit assignments.

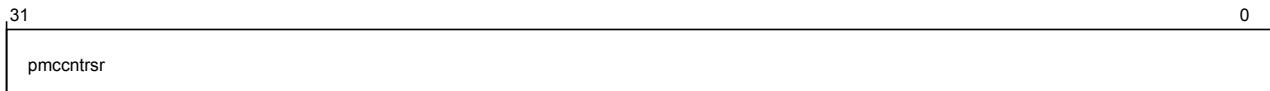


Figure 3-882 por_dt_por_dt_pmccntrs (low)

The following table shows the por_dt_pmccntrs lower register bit assignments.

Table 3-902 por_dt_por_dt_pmccntrs (low)

Bits	Field name	Description	Type	Reset
31:0	pmccntrs	PMU cycle counter shadow register	RW	40'h0

por_dt_pmcr

Functions as the PMU control register.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2100
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

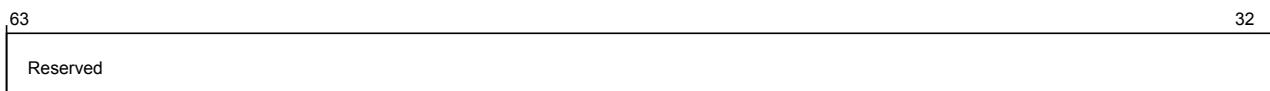


Figure 3-883 por_dt_por_dt_pmcr (high)

The following table shows the por_dt_pmcr higher register bit assignments.

Table 3-903 por_dt_por_dt_pmcr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

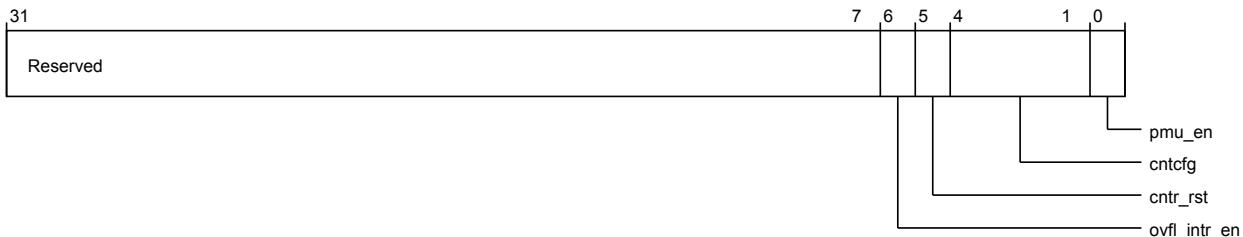


Figure 3-884 por_dt_por_dt_pmcr (low)

The following table shows the por_dt_pmcr lower register bit assignments.

Table 3-904 por_dt_por_dt_pmcr (low)

Bits	Field name	Description	Type	Reset
31:7	Reserved	Reserved	RO	-
6	ovfl_intr_en	Enables INTREQPMU assertion on PMU counter overflow	RW	1'h0
5	cntr_rst	Enables clearing of live counters upon assertion of por_dt_pmsrr.ss_req or PMUSNAPSHOTREQ	RW	1'h0
4:1	cntcfg	Groups adjacent 32-bit registers into a 64-bit register	RW	4'h0
0	pmu_en	Enables PMU features	RW	1'b0

por_dt_pmovsr

Provides the PMU overflow status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h2118

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	Reserved	32
----	----------	----

Figure 3-885 por_dt_por_dt_pmovsr (high)

The following table shows the por_dt_pmovsr higher register bit assignments.

Table 3-905 por_dt_por_dt_pmovsr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31	Reserved	9	8	0
				pmovsr

Figure 3-886 por_dt_por_dt_pmovsr (low)

The following table shows the por_dt_pmovsr lower register bit assignments.

Table 3-906 por_dt_por_dt_pmovsr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8:0	pmovsr	PMU overflow status Bit 8: Indicates overflow from cycle counter Bits [7:0]: Indicates overflow from counters 7 to 0	RO	9'h0

por_dt_pmovsr_clr

Clears the PMU overflow status.

Its characteristics are:

Type WO

Register width (Bits) 64

Address offset 16'h2120

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

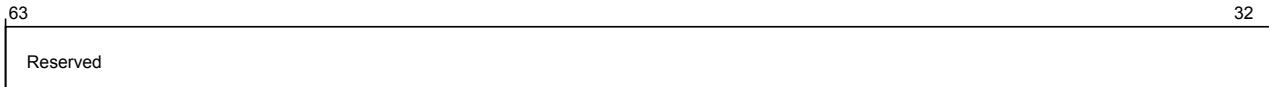


Figure 3-887 por_dt_por_dt_pmovsr_clr (high)

The following table shows the por_dt_pmovsr_clr higher register bit assignments.

Table 3-907 por_dt_por_dt_pmovsr_clr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

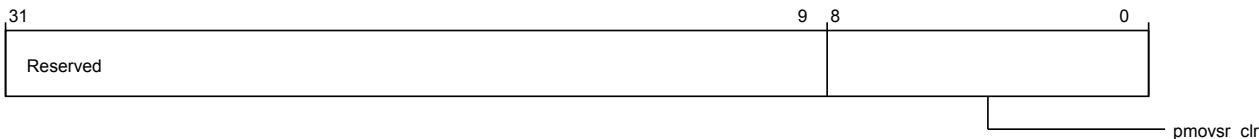


Figure 3-888 por_dt_por_dt_pmovsr_clr (low)

The following table shows the por_dt_pmovsr_clr lower register bit assignments.

Table 3-908 por_dt_por_dt_pmovsr_clr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8:0	pmovsr_clr	Write a 1 to clear the corresponding bit in por_dt_pmovsr.pmovsr	WO	9'b0

por_dt_pmssr

Provides the PMU snapshot status.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h2128

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

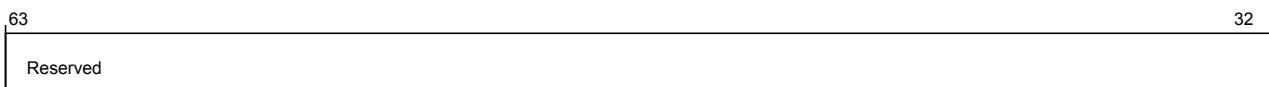


Figure 3-889 por_dt_por_dt_pmssr (high)

The following table shows the por_dt_pmssr higher register bit assignments.

Table 3-909 por_dt_por_dt_pmssr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

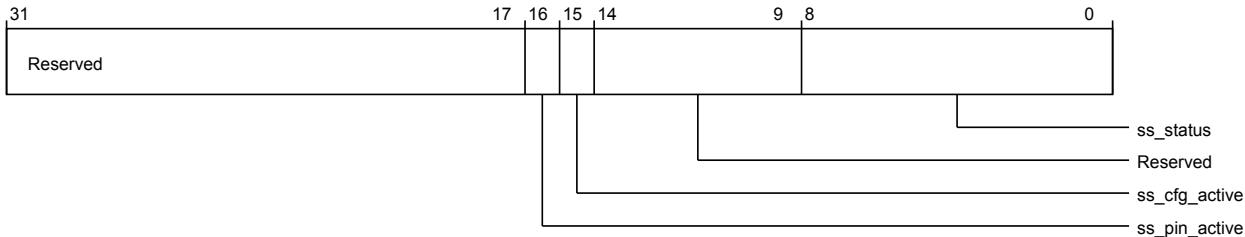


Figure 3-890 por_dt_por_dt_pmssr (low)

The following table shows the por_dt_pmssr lower register bit assignments.

Table 3-910 por_dt_por_dt_pmssr (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	ss_pin_active	Activates PMU snapshot from PMUSNAPSHOTREQ	RO	1'b0
15	ss_cfg_active	PMU snapshot activated from configuration write	RO	1'b0
14:9	Reserved	Reserved	RO	-
8:0	ss_status	PMU snapshot status Bit 8: Indicates snapshot status for cycle counter Bits [7:0]: Indicates snapshot status for counters 7 to 0	RO	9'b0

por_dt_pmsrr

Sends PMU snapshot requests.

Its characteristics are:

Type	WO
Register width (Bits)	64
Address offset	16'h2130
Register reset	64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

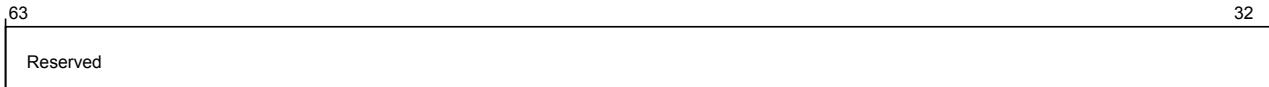


Figure 3-891 por_dt_por_dt_pmsrr (high)

The following table shows the por_dt_pmsrr higher register bit assignments.

Table 3-911 por_dt_por_dt_pmsrr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

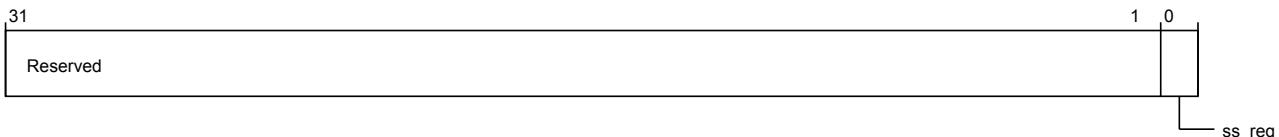


Figure 3-892 por_dt_por_dt_pmsrr (low)

The following table shows the port pmsrr lower register bit assignments.

Table 3-912 por_dt_por_dt_pmsrr (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	ss_req	Write a 1 to request PMU snapshot	WO	1'b0

por_dt_claim

Functions as the claim tag set register.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16 hFA0

Usage constraints There are no usage constraints

The following image shows the higher register bit assignments

Figure 3-893 por dt por dt claim (high)

The following table shows the por_dt_claim higher register bit assignments.

Table 3-913 por_dt_por_dt_claim (high)

Bits	Field name	Description	Type	Reset
63:32	clr	Upper half of the claim tag value; enables individual bits to be cleared (write) and returns the current claim tag value (read)	RW	32'b0

The following image shows the lower register bit assignments.



Figure 3-894 por_dt_por_dt_claim (low)

The following table shows the por_dt_claim lower register bit assignments.

Table 3-914 por_dt_por_dt_claim (low)

Bits	Field name	Description	Type	Reset
31:0	set	Lower half of the claim tag value; allows individual bits to be set (write) and returns the number of bits that can be set (read)	RW	32'hffffffff

por_dt_devaff

Functions as the device affinity register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFA8

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

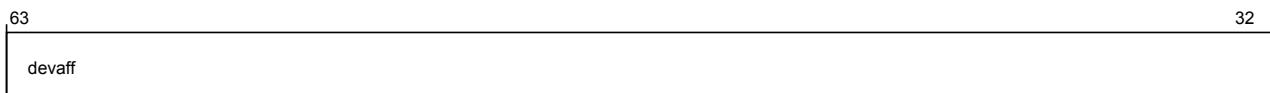


Figure 3-895 por_dt_por_dt_devaff (high)

The following table shows the por_dt_devaff higher register bit assignments.

Table 3-915 por_dt_por_dt_devaff (high)

Bits	Field name	Description	Type	Reset
63:32	devaff	Device affinity register	RO	64'b0

The following image shows the lower register bit assignments.

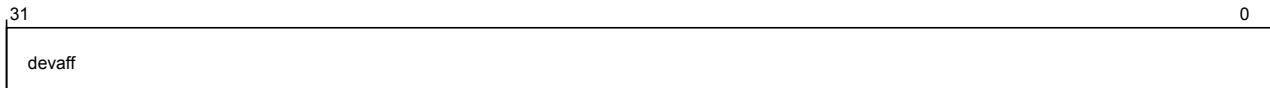


Figure 3-896 por_dt_por_dt_devaff (low)

The following table shows the por_dt_devaff lower register bit assignments.

Table 3-916 por_dt_por_dt_devaff (low)

Bits	Field name	Description	Type	Reset
31:0	devaff	Device affinity register	RO	64'b0

por_dt_lsr

Functions as the lock status register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFB0

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

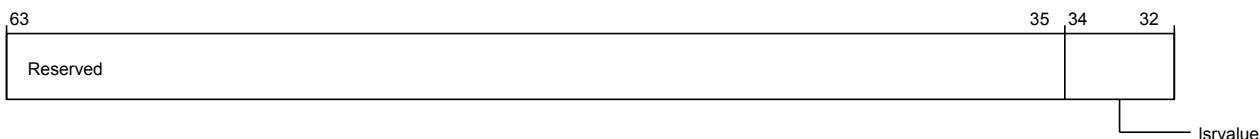


Figure 3-897 por_dt_por_dt_lsr (high)

The following table shows the por_dt_lsr higher register bit assignments.

Table 3-917 por_dt_por_dt_lsr (high)

Bits	Field name	Description	Type	Reset
63:35	Reserved	Reserved	RO	-
34:32	lsrvalue	Lock status value	RO	3'b0

The following image shows the lower register bit assignments.

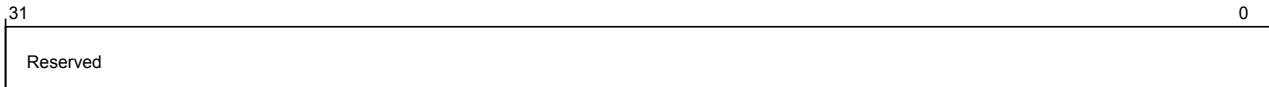


Figure 3-898 por_dt_por_dt_lsr (low)

The following table shows the por_dt_lsr lower register bit assignments.

Table 3-918 por_dt_por_dt_lsr (low)

Bits	Field name	Description	Type	Reset
31:0	Reserved	Reserved	RO	-

por_dt authstatus devarch

Functions as the authentication status register and the device architecture register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFB8

Register reset 64'b01001010

Usage constraints There are no usage constraints.

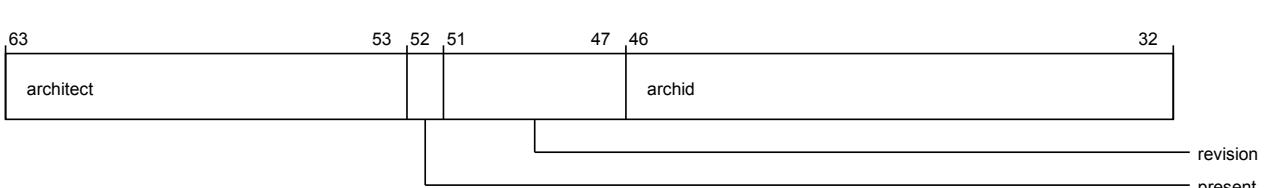


Figure 3-899 por dt por dt authstatus devarch (high)

The following table shows the port authstatus devarch higher register bit assignments.

Table 3-919 por dt por dt authstatus devarch (high)

Bits	Field name	Description	Type	Reset
63:53	architect	Architect	RO	11'b0
52	present	Present	RO	1'b1
51:47	revision	Architecture revision	RO	6'b0
46:32	archid	Architecture ID	RO	16'b0

The following image shows the lower register bit assignments.

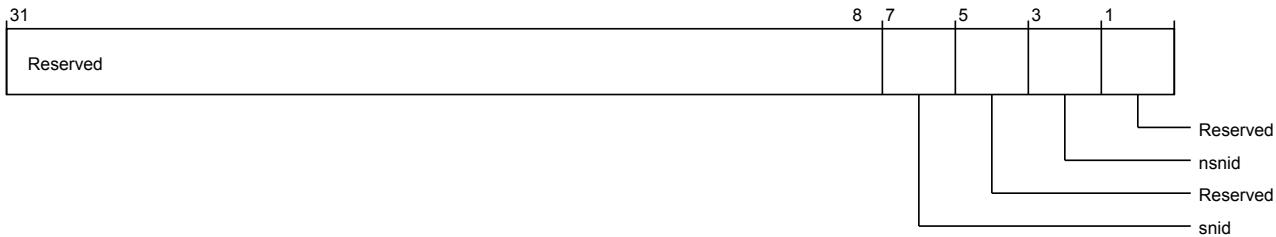


Figure 3-900 por_dt_por_dt_authstatus_devarch (low)

The following table shows the por_dt_authstatus_devarch lower register bit assignments.

Table 3-920 por_dt_por_dt_authstatus_devarch (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:6	snid	Secure non-invasive debug	RO	2'b10
5:4	Reserved	Reserved	RO	-
3:2	nsnid	Non-secure non-invasive debug	RO	2'b10
1:0	Reserved	Reserved	RO	-

por_dt_devid

Functions as the device configuration register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFC0

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

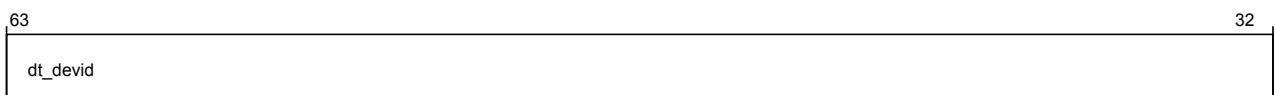


Figure 3-901 por_dt_por_dt_devid (high)

The following table shows the por_dt_devid higher register bit assignments.

Table 3-921 por_dt_por_dt_devid (high)

Bits	Field name	Description	Type	Reset
63:32	dt_devid	Device ID	RO	64'b0

The following image shows the lower register bit assignments.

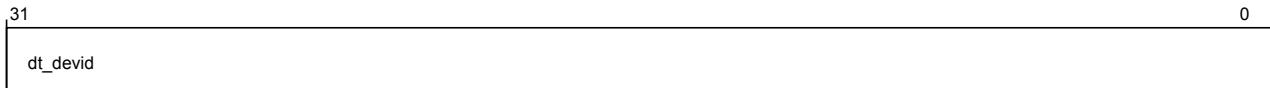


Figure 3-902 por_dt_por_dt_devid (low)

The following table shows the por_dt_devid lower register bit assignments.

Table 3-922 por_dt_por_dt_devid (low)

Bits	Field name	Description	Type	Reset
31:0	dt_devid	Device ID	RO	64'b0

por_dt_devtype

Functions as the device type identifier register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFC8

Register reset 64'b01000011

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

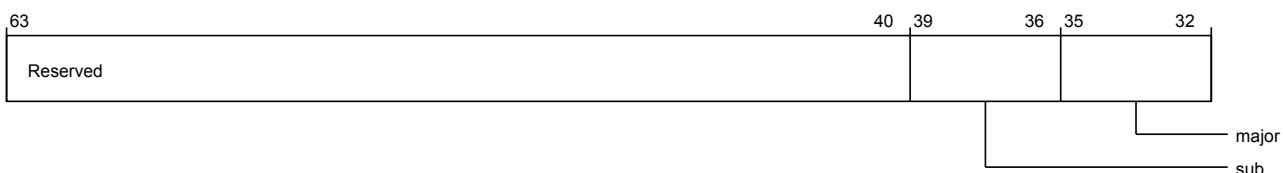


Figure 3-903 por_dt_por_dt_devtype (high)

The following table shows the por_dt_devtype higher register bit assignments.

Table 3-923 por_dt_por_dt_devtype (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:36	sub	Sub type	RO	4'h4
35:32	major	Major type	RO	4'h3

The following image shows the lower register bit assignments.

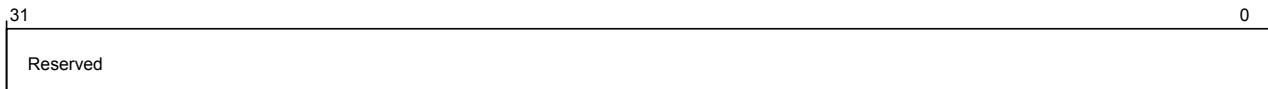


Figure 3-904 por_dt_por_dt_devtype (low)

The following table shows the por_dt_devtype lower register bit assignments.

Table 3-924 por_dt_por_dt_devtype (low)

Bits	Field name	Description	Type	Reset
31:0	Reserved	Reserved	RO	-

por_dt_pidr45

Functions as the identification register for peripheral ID 4 and peripheral ID 5.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFD0

Register reset 64'b000000100

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

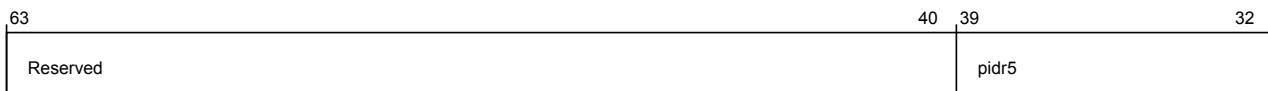


Figure 3-905 por_dt_por_dt_pidr45 (high)

The following table shows the por_dt_pidr45 higher register bit assignments.

Table 3-925 por_dt_por_dt_pidr45 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr5	Peripheral ID 5	RO	8'b0

The following image shows the lower register bit assignments.

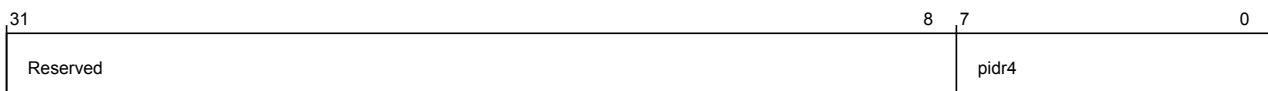


Figure 3-906 por_dt_por_dt_pidr45 (low)

The following table shows the por_dt_pidr45 lower register bit assignments.

Table 3-926 por_dt_por_dt_pidr45 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr4	Peripheral ID 4	RO	8'h4

por_dt_pidr67

Functions as the identification register for peripheral ID 6 and peripheral ID 7.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFD8

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

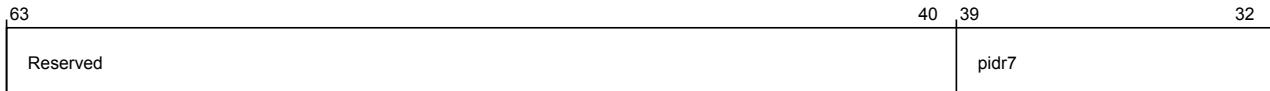


Figure 3-907 por_dt_por_dt_pidr67 (high)

The following table shows the por_dt_pidr67 higher register bit assignments.

Table 3-927 por_dt_por_dt_pidr67 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr7	Peripheral ID 7	RO	8'b0

The following image shows the lower register bit assignments.

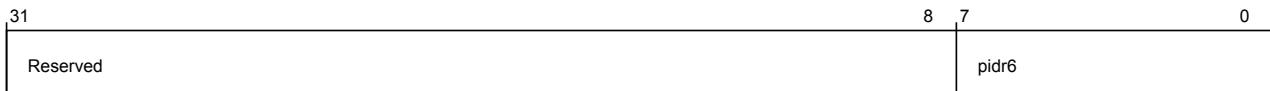


Figure 3-908 por_dt_por_dt_pidr67 (low)

The following table shows the por_dt_pidr67 lower register bit assignments.

Table 3-928 por_dt_por_dt_pidr67 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr6	Peripheral ID 6	RO	8'b0

por_dt_pidr01

Functions as the identification register for peripheral ID 0 and peripheral ID 1.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFE0

Register reset 64'b0101110000011100

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-909 por_dt_por_dt_pidr01 (high)

The following table shows the por_dt_pidr01 higher register bit assignments.

Table 3-929 por_dt_por_dt_pidr01 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr1	Peripheral ID 1	RO	8'hb4

The following image shows the lower register bit assignments.

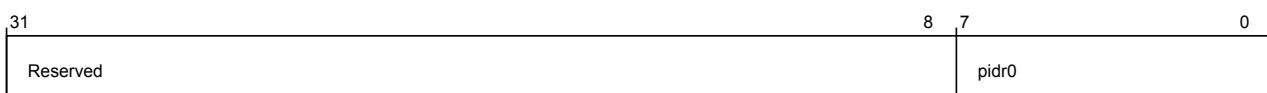


Figure 3-910 por_dt_por_dt_pidr01 (low)

The following table shows the por_dt_pidr01 lower register bit assignments.

Table 3-930 por_dt_por_dt_pidr01 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr0	Peripheral ID 0	RO	8'h34

por_dt_pidr23

Functions as the identification register for peripheral ID 2 and peripheral ID 3.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFE8

Register reset 64'b0000000111

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-911 por_dt_por_dt_pidr23 (high)

The following table shows the por_dt_pidr23 higher register bit assignments.

Table 3-931 por_dt_por_dt_pidr23 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pidr3	Peripheral ID 3	RO	8'b0

The following image shows the lower register bit assignments.

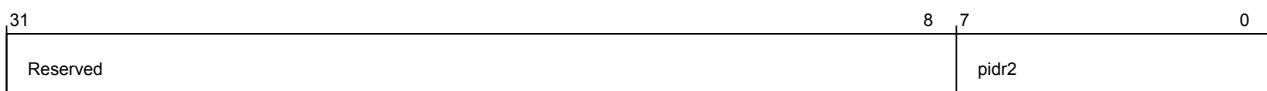


Figure 3-912 por_dt_por_dt_pidr23 (low)

The following table shows the por_dt_pidr23 lower register bit assignments.

Table 3-932 por_dt_por_dt_pidr23 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	pidr2	Peripheral ID 2	RO	8'h7

por_dt_cidr01

Functions as the identification register for component ID 0 and component ID 1.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFF0

Register reset 64'b100111100001101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-913 por_dt_por_dt_cidr01 (high)

The following table shows the por_dt_cidr01 higher register bit assignments.

Table 3-933 por_dt_por_dt_cidr01 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	cidr1	Component ID 1	RO	8'h9f

The following image shows the lower register bit assignments.

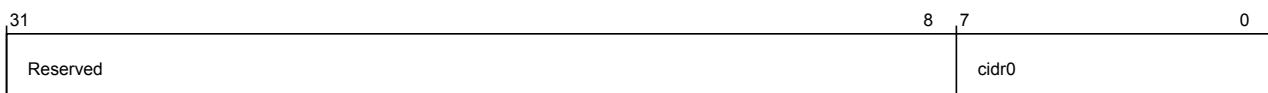


Figure 3-914 por_dt_por_dt_cidr01 (low)

The following table shows the por_dt_cidr01 lower register bit assignments.

Table 3-934 por_dt_por_dt_cidr01 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	cidr0	Component ID 0	RO	8'hd

por_dt_cidr23

Functions as the identification register for component ID 2 and component ID 3.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hFF8

Register reset 64'b0001011100000101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-915 por_dt_por_dt_cidr23 (high)

The following table shows the por_dt_cidr23 higher register bit assignments.

Table 3-935 por_dt_por_dt_cidr23 (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	cidr3	Component ID 3	RO	8'hb1

The following image shows the lower register bit assignments.



Figure 3-916 por_dt_por_dt_cidr23 (low)

The following table shows the por_dt_cidr23 lower register bit assignments.

Table 3-936 por_dt_por_dt_cidr23 (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:0	cidr2	Component ID 2	RO	8'h5

3.3.15 SBSX register descriptions

This section lists the SBSX registers.

por_sbsx_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

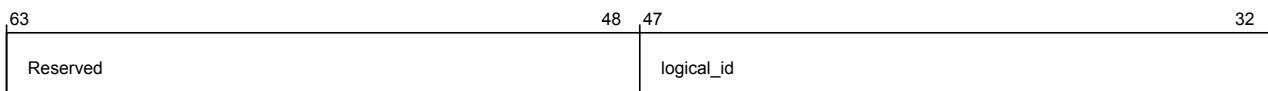


Figure 3-917 por_sbsx_por_sbsx_node_info (high)

The following table shows the por_sbsx_node_info higher register bit assignments.

Table 3-937 por_sbsx_por_sbsx_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

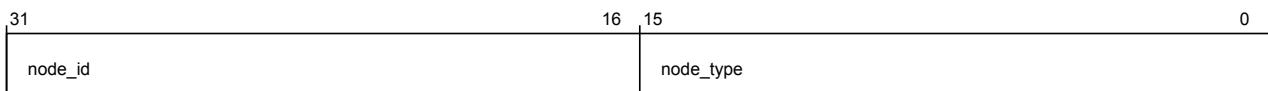


Figure 3-918 por_sbsx_por_sbsx_node_info (low)

The following table shows the por_sbsx_node_info lower register bit assignments.

Table 3-938 por_sbsx_por_sbsx_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0007

por_sbsx_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

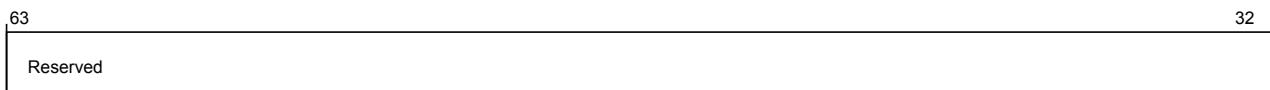


Figure 3-919 por_sbsx_por_sbsx_child_info (high)

The following table shows the por_sbsx_child_info higher register bit assignments.

Table 3-939 por_sbsx_por_sbsx_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

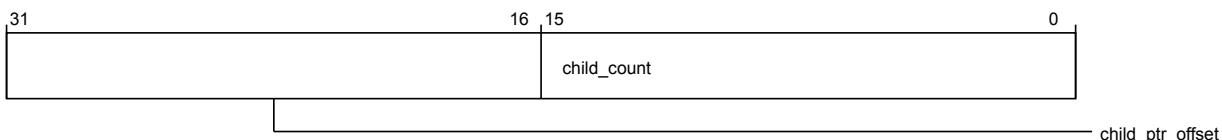


Figure 3-920 por_sbsx_por_sbsx_child_info (low)

The following table shows the por_sbsx_child_info lower register bit assignments.

Table 3-940 por_sbsx_por_sbsx_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_sbsx_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h980
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

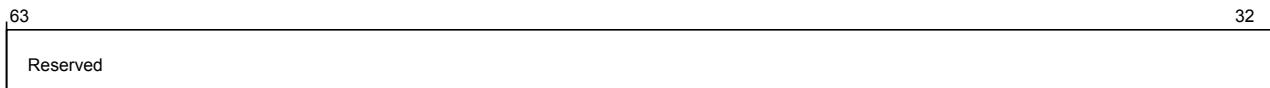


Figure 3-921 por_sbsx_por_sbsx_secure_register_groups_override (high)

The following table shows the por_sbsx_secure_register_groups_override higher register bit assignments.

Table 3-941 por_sbsx_por_sbsx_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

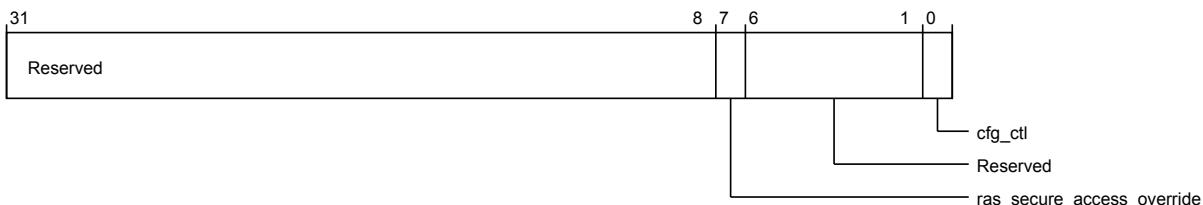


Figure 3-922 por_sbsx_por_sbsx_secure_register_groups_override (low)

The following table shows the por_sbsx_secure_register_groups_override lower register bit assignments.

Table 3-942 por_sbsx_por_sbsx_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0

Table 3-942 por_sbsx_por_sbsx_secure_register_groups_override (low) (continued)

Bits	Field name	Description	Type	Reset
6:1	Reserved	Reserved	RO	-
0	cfg_ctl	Allows non-secure access to secure configuration control register (por_sbsx_cfg_ctl)	RW	1'b0

por_sbsx_unit_info

Provides component identification information for SBSX.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

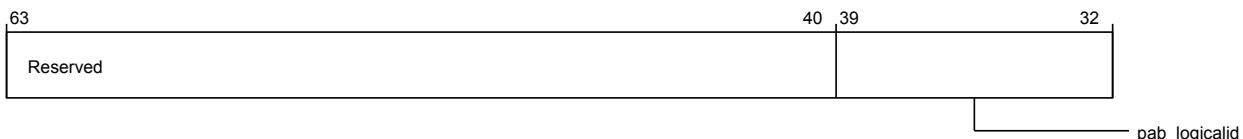


Figure 3-923 por_sbsx_por_sbsx_unit_info (high)

The following table shows the por_sbsx_unit_info higher register bit assignments.

Table 3-943 por_sbsx_por_sbsx_unit_info (high)

Bits	Field name	Description	Type	Reset
63:40	Reserved	Reserved	RO	-
39:32	pab_logicalid	PUB AUB bridge Logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

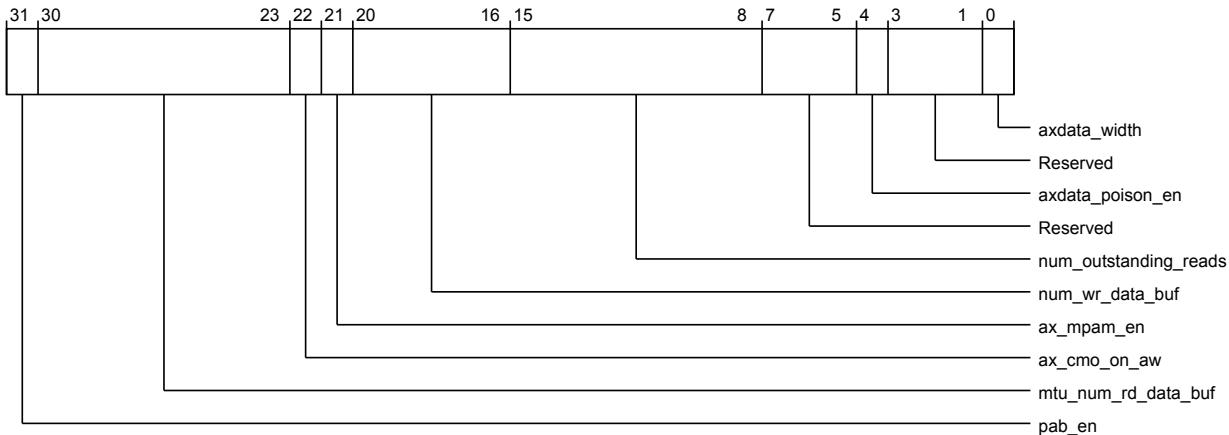


Figure 3-924 por_sbsx_por_sbsx_unit_info (low)

The following table shows the por_sbsx_unit_info lower register bit assignments.

Table 3-944 por_sbsx_por_sbsx_unit_info (low)

Bits	Field name	Description	Type	Reset
31	pab_en	PUB AUB bridge enable 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
30:23	mtu_num_rd_data_buf	Number of mtu read data buffers in SBSX	RO	Configuration dependent
22	ax_cmo_on_aw	Write Channel CMOs enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
21	ax_mpam_en	MPAM enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
20:16	num_wr_data_buf	Number of write data buffers in SBSX	RO	Configuration dependent
15:8	num_outstanding_reads	Maximum number of outstanding AXI read requests from SBSX	RO	Configuration dependent
7:5	Reserved	Reserved	RO	-
4	axdata_poison_en	Data poison support on ACE-Lite/AXI4 interface 1'b0: Not supported 1'b1: Supported	RO	Configuration dependent
3:1	Reserved	Reserved	RO	-
0	axdata_width	Data width on ACE-Lite/AXI4 interface 1'b0: 128 bits 1'b1: 256 bits	RO	Configuration dependent

por_sbsx_cfg_ctl

Functions as the configuration control register for SBSX bridge.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_sbsx_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.



Figure 3-925 por_sbsx_por_sbsx_cfg_ctl (high)

The following table shows the por_sbsx_cfg_ctl higher register bit assignments.

Table 3-945 por_sbsx_por_sbsx_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

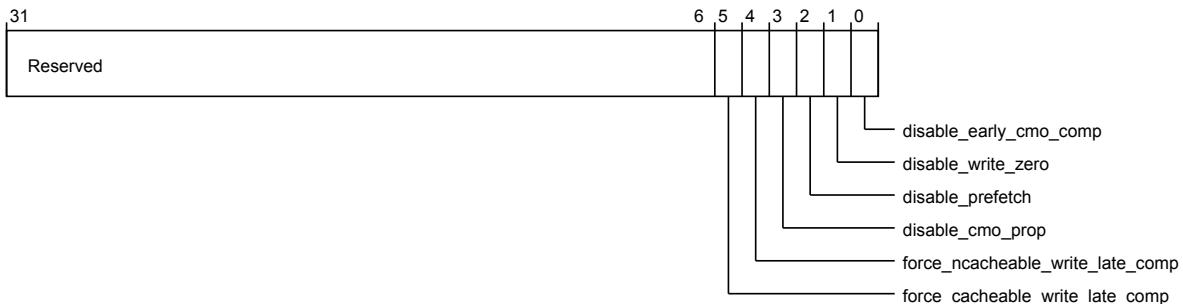


Figure 3-926 por_sbsx_por_sbsx_cfg_ctl (low)

The following table shows the por_sbsx_cfg_ctl lower register bit assignments.

Table 3-946 por_sbsx_por_sbsx_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5	force_cacheable_write_late_comp	Late Comp for Cacheable Writes. Overrides EWA	RW	1'b0

Table 3-946 por_sbsx_por_sbsx_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
4	force_ncacheable_write_late_comp	Late Comp for Non-cacheable Writes. Overrides EWA	RW	1'b0
3	disable_cmo_prop	Disables CMO propagation on ACE.	RW	1'b0
2	disable_prefetch	Disables Prefetches on AXI.	RW	1'b0
1	disable_write_zero	Disables WriteZero Op on AXI.	RW	1'b0
0	disable_early_cmo_comp	Disables Early Comp for CMOs in SBSX to HNF.	RW	1'b0

por_sbsx_aux_ctl

Functions as the auxiliary control register for the SBSX bridge.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA08

Register reset 64'b0

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

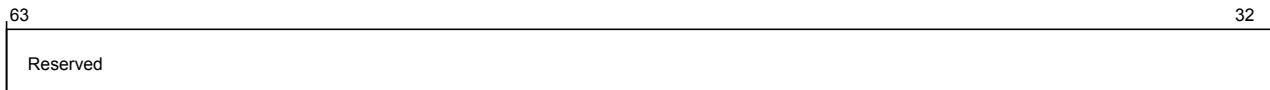


Figure 3-927 por_sbsx_por_sbsx_aux_ctl (high)

The following table shows the por_sbsx_aux_ctl higher register bit assignments.

Table 3-947 por_sbsx_por_sbsx_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

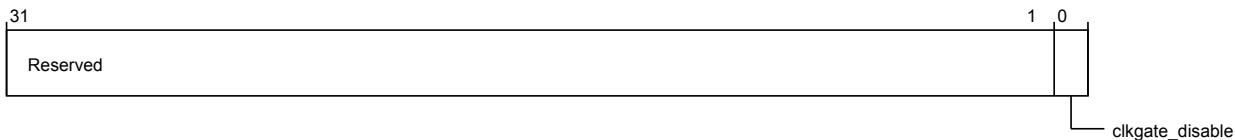


Figure 3-928 por_sbsx_por_sbsx_aux_ctl (low)

The following table shows the por_sbsx_aux_ctl lower register bit assignments.

Table 3-948 por_sbsx_por_sbsx_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	clkgate_disable	Disables internal clock gating in SBSX bridge	RW	1'b0

por_sbsx_cbusy_limit_ctl

Cbusy threshold limits for Request Tracker entries.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA18

Register reset Configuration dependent

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

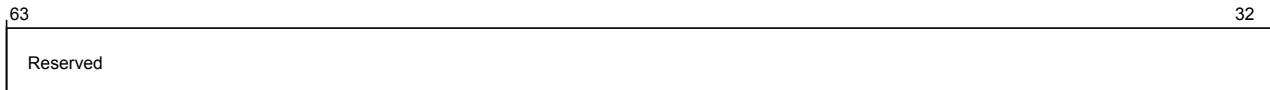


Figure 3-929 por_sbsx_por_sbsx_cbusy_limit_ctl (high)

The following table shows the por_sbsx_cbusy_limit_ctl higher register bit assignments.

Table 3-949 por_sbsx_por_sbsx_cbusy_limit_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

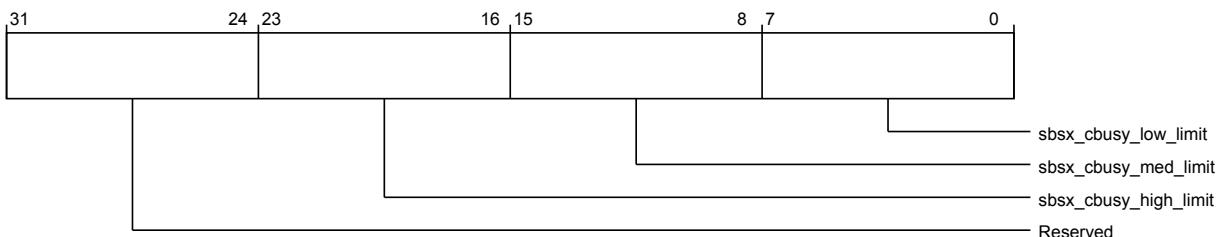


Figure 3-930 por_sbsx_por_sbsx_cbusy_limit_ctl (low)

The following table shows the por_sbsx_cbusy_limit_ctl lower register bit assignments.

Table 3-950 por_sbsx_por_sbsx_cbusy_limit_ctl (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	sbsx_cbusy_high_limit	ReqTracker limit for CBusy High	RW	Configuration dependent
15:8	sbsx_cbusy_med_limit	ReqTracker limit for CBusy Med	RW	Configuration dependent
7:0	sbsx_cbusy_low_limit	ReqTracker limit for CBusy Low	RW	Configuration dependent

por_sbsx_errfr

Functions as the error feature register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3000

Register reset 64'b0000010100001

Usage constraints Only accessible by secure accesses.

Secure group override por_sbsx_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-931 por_sbsx_por_sbsx_errfr (high)

The following table shows the por_sbsx_errfr higher register bit assignments.

Table 3-951 por_sbsx_por_sbsx_errfr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

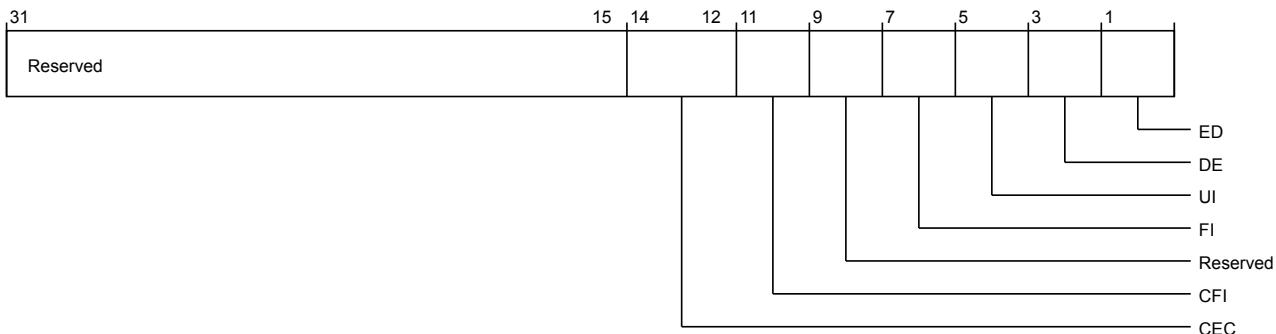


Figure 3-932 por_sbsx_por_sbsx_errfr (low)

The following table shows the por sbsx errfr lower register bit assignments.

Table 3-952 por_sbsx_por_sbsx_errfr (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors	RO	2'b00
1:0	ED	Error detection	RO	2'b01

por_sbsx_errctlr

Functions as the error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3008
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_sbsx_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

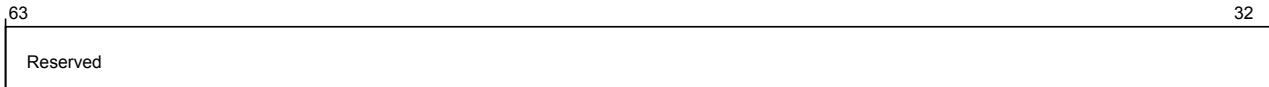


Figure 3-933 por_sbsx_por_sbsx_errctlr (high)

The following table shows the por_sbsx_errctlr higher register bit assignments.

Table 3-953 por_sbsx_por_sbsx_errctlr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

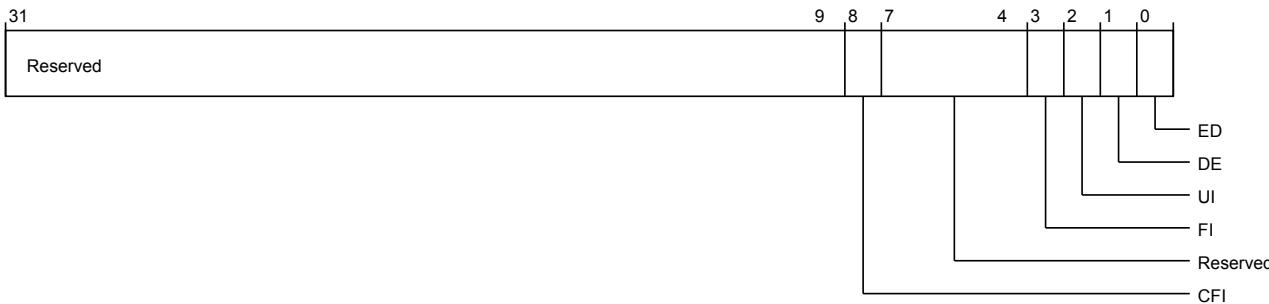


Figure 3-934 por_sbsx_por_sbsx_errctlr (low)

The following table shows the por_sbsx_errctlr lower register bit assignments.

Table 3-954 por_sbsx_por_sbsx_errctlr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_sbsx_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_sbsx_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_sbsx_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_sbsx_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_sbsx_errfr.ED	RW	1'b0

por_sbsx_errstatus

Functions as the error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h3010

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_sbsx_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-935 por_sbsx_por_sbsx_errstatus (high)

The following table shows the por_sbsx_errstatus higher register bit assignments.

Table 3-955 por_sbsx_por_sbsx_errstatus (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

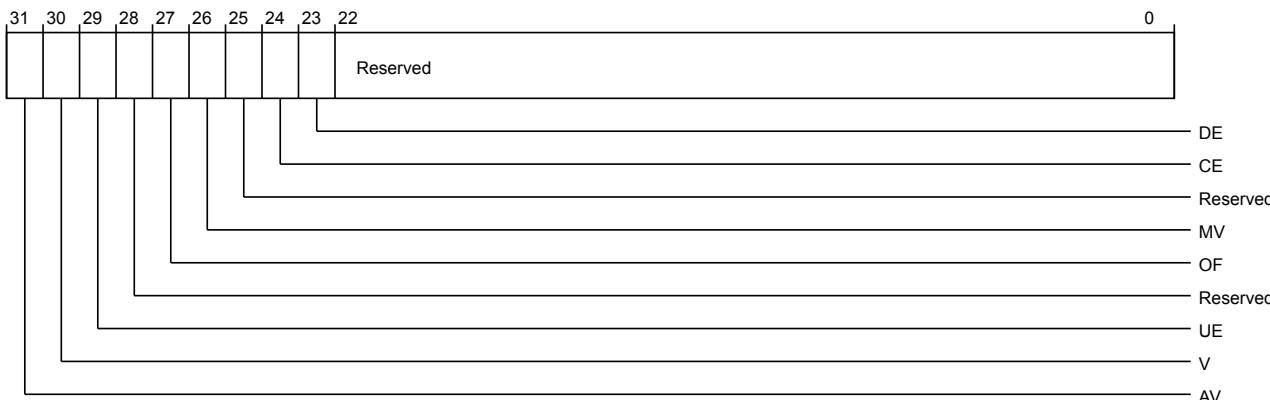


Figure 3-936 por_sbsx_por_sbsx_errstatus (low)

The following table shows the por_sbsx_errstatus lower register bit assignments.

Table 3-956 por_sbsx_por_sbsx_errstatus (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_sbsx_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_sbsx_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_sbsx_erraddr

Contains the error record address.

Its characteristics are:

Type	RW
------	----

Register width (Bits)	64
Address offset	16'h3018
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_sbsx_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

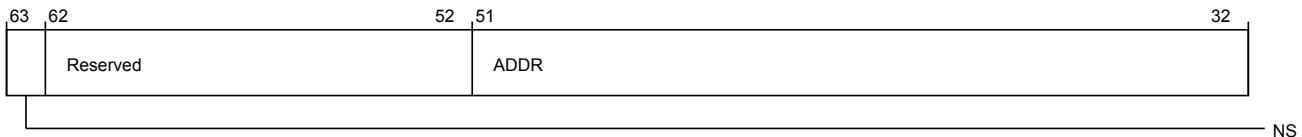


Figure 3-937 por_sbsx_por_sbsx_erraddr (high)

The following table shows the por_sbsx_erraddr higher register bit assignments.

Table 3-957 por_sbsx_por_sbsx_erraddr (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_sbsx_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.



Figure 3-938 por_sbsx_por_sbsx_erraddr (low)

The following table shows the por_sbsx_erraddr lower register bit assignments.

Table 3-958 por_sbsx_por_sbsx_erraddr (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_sbsx_errmisc

Functions as the miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3020
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_sbsx_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-939 por_sbsx_por_sbsx_errmisc (high)

The following table shows the por sbxx errmisc higher register bit assignments.

Table 3-959 por_sbsx_por_sbsx_errmisc (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

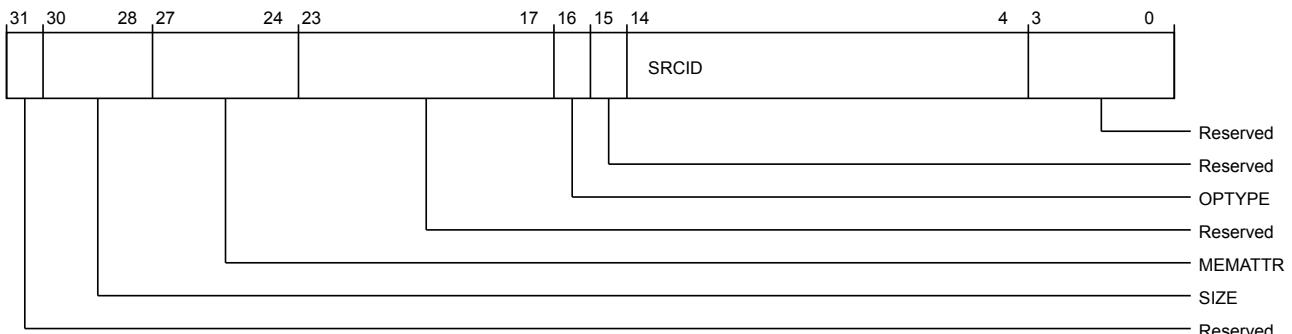


Figure 3-940 por_sbsx_por_sbsx_errmisc (low)

The following table shows the portability register bit assignments.

Table 3-960 por_sbsx_por_sbsx_errmisc (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	SIZE	Error transaction size	RW	3'b0
27:24	MEMATTR	Error memory attributes	RW	4'b0
23:17	Reserved	Reserved	RO	-
16	OPTYPE	Error opcode type 1'b1: WR_NO_SNP_PTL (partial) 1'b0: WR_NO_SNP_FULL	RW	1'b0
15	Reserved	Reserved	RO	-
14:4	SRCID	Error source ID	RW	11'b0
3:0	Reserved	Reserved	RO	-

por_sbsx_errfr_NS

Functions as the non-secure error feature register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3100

Register reset 64'b0000010100001

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

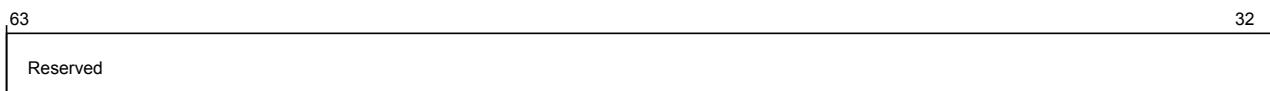


Figure 3-941 por_sbsx_por_sbsx_errfr_ns (high)

The following table shows the por_sbsx_errfr_NS higher register bit assignments.

Table 3-961 por_sbsx_por_sbsx_errfr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

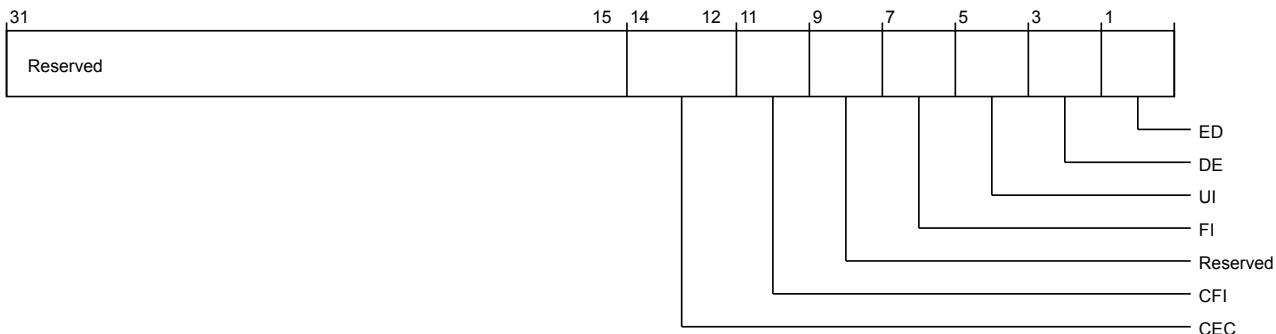


Figure 3-942 por_sbsx_por_sbsx_errfr_ns (low)

The following table shows the por sbsx errfr NS lower register bit assignments.

Table 3-962 por_sbsx_por_sbsx_errfr_ns (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors	RO	2'b00
1:0	ED	Error detection	RO	2'b01

por_sbsx_errctlr_NS

Functions as the non-secure error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3

Register reset 64'b0

Usage constraints There are no usage constraints.

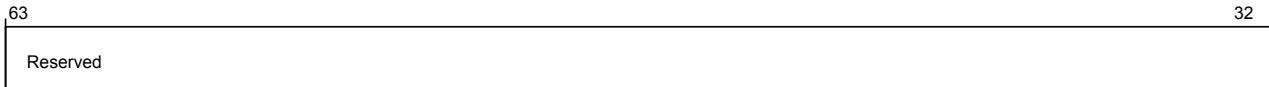


Figure 3-943 por_sbsx_por_sbsx_errctlr_ns (high)

The following table shows the por_sbsx_errctlr_NS higher register bit assignments.

Table 3-963 por_sbsx_por_sbsx_errctlr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

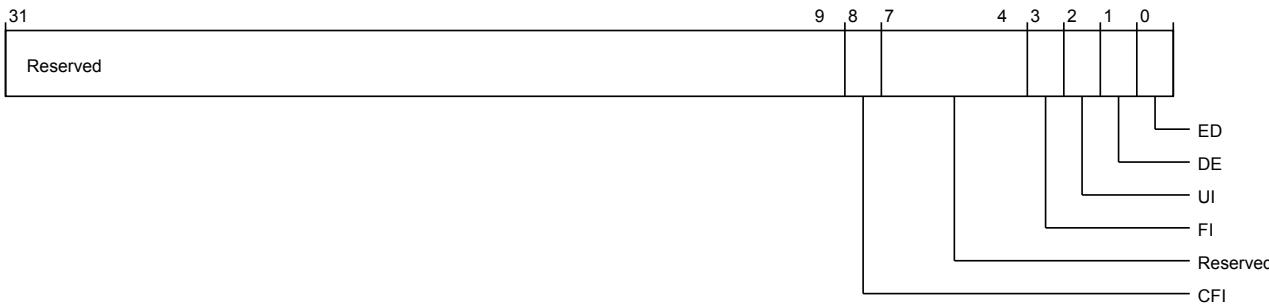


Figure 3-944 por_sbsx_por_sbsx_errctlr_ns (low)

The following table shows the por_sbsx_errctlr_NS lower register bit assignments.

Table 3-964 por_sbsx_por_sbsx_errctlr_ns (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_sbsx_errfr_NS.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_sbsx_errfr_NS.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_sbsx_errfr_NS.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_sbsx_errfr_NS.DE	RW	1'b0
0	ED	Enables error detection as specified in por_sbsx_errfr_NS.ED	RW	1'b0

por_sbsx_errstatus_NS

Functions as the non-secure error status register.

Its characteristics are:

Type W1C

Register width (Bits)	64
Address offset	16'h3110
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

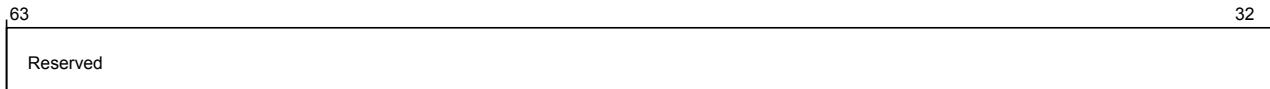


Figure 3-945 por_sbsx_por_sbsx_errstatus_ns (high)

The following table shows the por_sbsx_errstatus_NS higher register bit assignments.

Table 3-965 por_sbsx_por_sbsx_errstatus_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

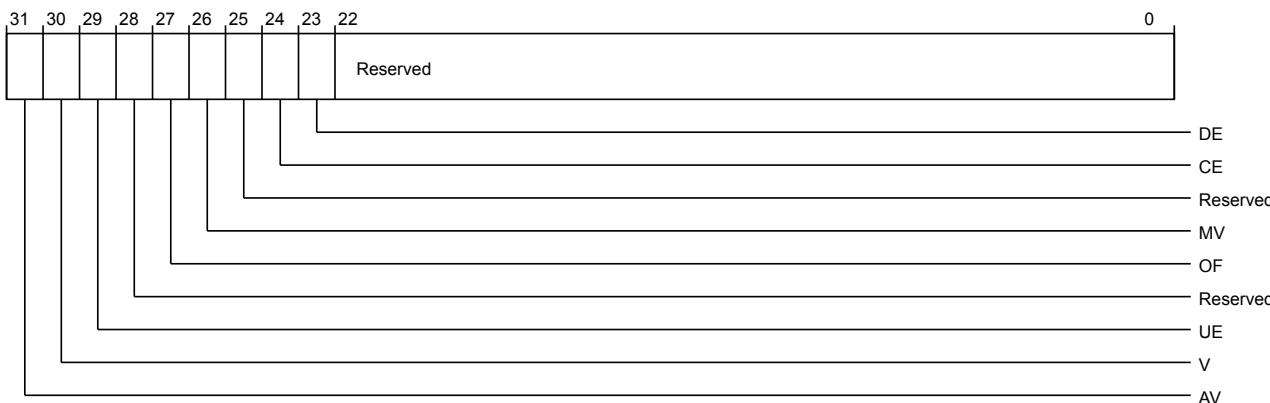


Figure 3-946 por_sbsx_por_sbsx_errstatus_ns (low)

The following table shows the por_sbsx_errstatus_NS lower register bit assignments.

Table 3-966 por_sbsx_por_sbsx_errstatus_ns (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_sbsx_erraddr_NS contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_sbsx_errmisc_NS valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_sbsx_erraddr_NS

Contains the non-secure error record address.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3118
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

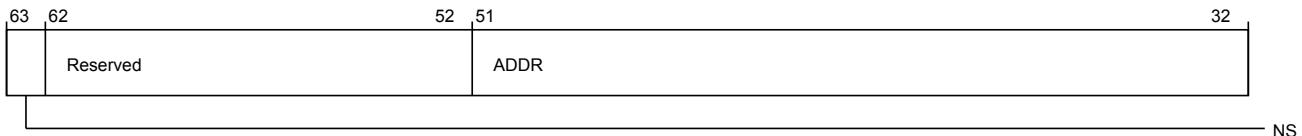


Figure 3-947 por_sbsx_por_sbsx_erraddr_ns (high)

The following table shows the por_sbsx_erraddr_NS higher register bit assignments.

Table 3-967 por_sbsx_por_sbsx_erraddr_ns (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_sbsx_erraddr_NS.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.



Figure 3-948 por_sbsx_por_sbsx_erraddr_ns (low)

The following table shows the por_sbsx_erraddr_NS lower register bit assignments.

Table 3-968 por_sbsx_por_sbsx_erraddr_ns (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_sbsx_errmisc_NS

Functions as the non-secure miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3120
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-949 por_sbsx_por_sbsx_errmisc_ns (high)

The following table shows the por_sbsx_errmisc_NS higher register bit assignments.

Table 3-969 por_sbsx_por_sbsx_errmisc_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

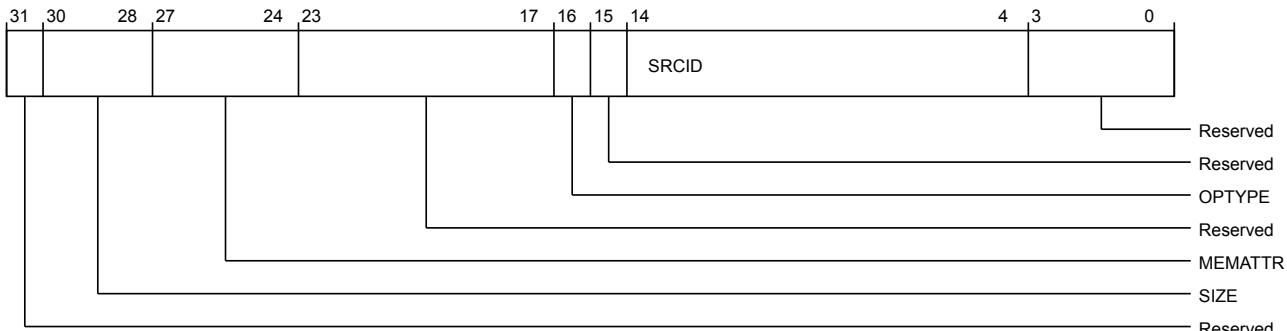


Figure 3-950 por_sbsx_por_sbsx_errmisc_ns (low)

The following table shows the por_sbsx_errmisc_NS lower register bit assignments.

Table 3-970 por_sbsx_por_sbsx_errmisc_ns (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	SIZE	Error transaction size	RW	3'b0
27:24	MEMATTR	Error memory attributes	RW	4'b0
23:17	Reserved	Reserved	RO	-

Table 3-970 por_sbsx_por_sbsx_errmisc_ns (low) (continued)

Bits	Field name	Description	Type	Reset
16	OPTYPE	Error opcode type 1'b1: WR_NO_SNP_PTL (partial) 1'b0: WR_NO_SNP_FULL	RW	1'b0
15	Reserved	Reserved	RO	-
14:4	SRCID	Error source ID	RW	11'b0
3:0	Reserved	Reserved	RO	-

por_sbsx_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

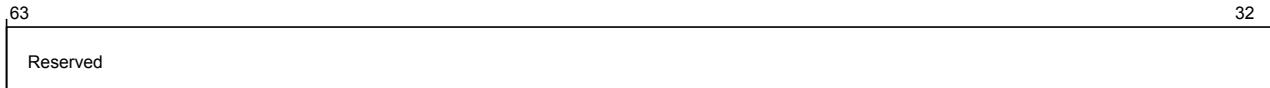


Figure 3-951 por_sbsx_por_sbsx_pmu_event_sel (high)

The following table shows the por_sbsx_pmu_event_sel higher register bit assignments.

Table 3-971 por_sbsx_por_sbsx_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

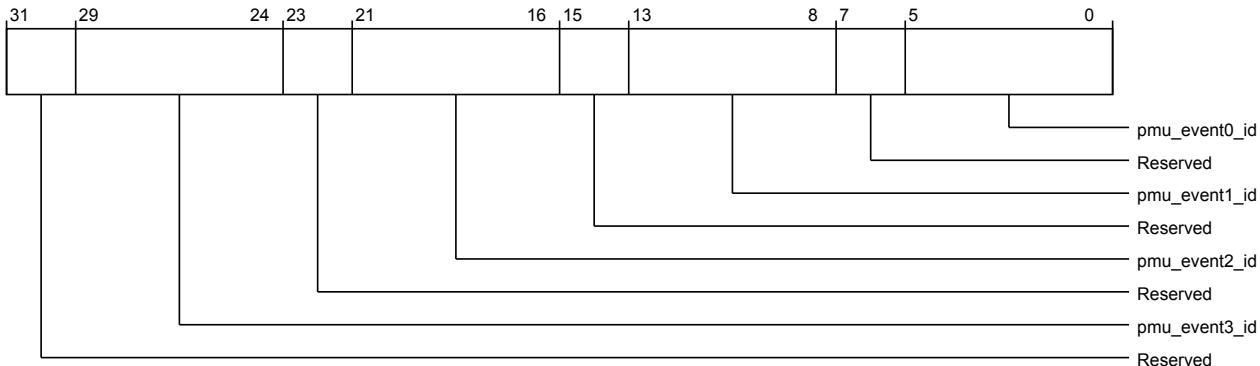


Figure 3-952 por_sbsx_por_sbsx_pmu_event_sel (low)

The following table shows the por_sbsx_pmu_event_sel lower register bit assignments.

Table 3-972 por_sbsx_por_sbsx_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	SBSX PMU Event 3 select; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	SBSX PMU Event 2 select; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	SBSX PMU Event 1 select; see pmu_event0_id for encodings	RW	6'b0

Table 3-972 por_sbsx_por_sbsx_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-
5:0	pmu_event0_id	SBSX PMU Event 0 select 6'h00: No event 6'h01: Read request 6'h02: Write request 6'h03: CMO request 6'h04: RETRYACK TXRSP flit sent 6'h05: TXDAT flit seen 6'h06: TXRSP flit seen 6'h11: Read request tracker occupancy count overflow 6'h12: Write request tracker occupancy count overflow 6'h13: CMO request tracker occupancy count overflow 6'h14: WDB occupancy count overflow 6'h15: Read AXI pending tracker occupancy count overflow 6'h16: CMO AXI pending tracker occupancy count overflow 6'h17: RDB occupancy count overflow. (Only when MTU is enabled) 6'h21: ARVALID set without ARREADY 6'h22: AWVALID set without AWREADY 6'h23: WVALID set without WREADY 6'h24: TXDAT stall (TXDAT valid but no link credit available) 6'h25: TXRSP stall (TXRSP valid but no link credit available) NOTE: All other encodings are reserved.	RW	6'b0

3.3.16 HN-F MPAM_NS register descriptions

This section lists the HN-F MPAM_NS registers.

por_hnf_mpam_ns_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	48 47	32
Reserved	logical_id	

Figure 3-953 por_hnf_mpam_ns_por_hnf_mpam_ns_node_info (high)

The following table shows the por_hnf_mpam_ns_node_info higher register bit assignments.

Table 3-973 por_hnf_mpam_ns_por_hnf_mpam_ns_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

31	16 15	0
node_id	node_type	

Figure 3-954 por_hnf_mpam_ns_por_hnf_mpam_ns_node_info (low)

The following table shows the por_hnf_mpam_ns_node_info lower register bit assignments.

Table 3-974 por_hnf_mpam_ns_por_hnf_mpam_ns_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0009

por_hnf_mpam_ns_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

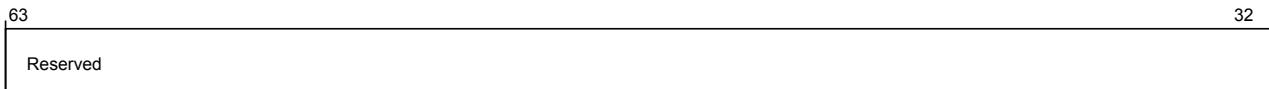


Figure 3-955 por_hnf_mpam_ns_por_hnf_mpam_ns_child_info (high)

The following table shows the por_hnf_mpam_ns_child_info higher register bit assignments.

Table 3-975 por_hnf_mpam_ns_por_hnf_mpam_ns_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

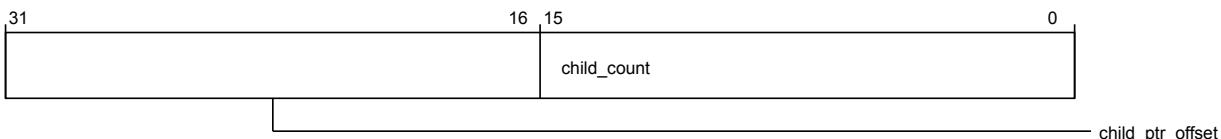


Figure 3-956 por_hnf_mpam_ns_por_hnf_mpam_ns_child_info (low)

The following table shows the por_hnf_mpam_ns_child_info lower register bit assignments.

Table 3-976 por_hnf_mpam_ns_por_hnf_mpam_ns_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_hnf_mpam_idr

MPAM features ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1000

Register reset Configuration dependent

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

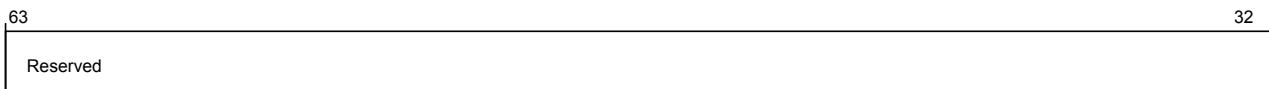


Figure 3-957 por_hnf_mpam_ns_por_hnf_mpam_idr (high)

The following table shows the por_hnf_mpam_idr higher register bit assignments.

Table 3-977 por_hnf_mpam_ns_por_hnf_mpam_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

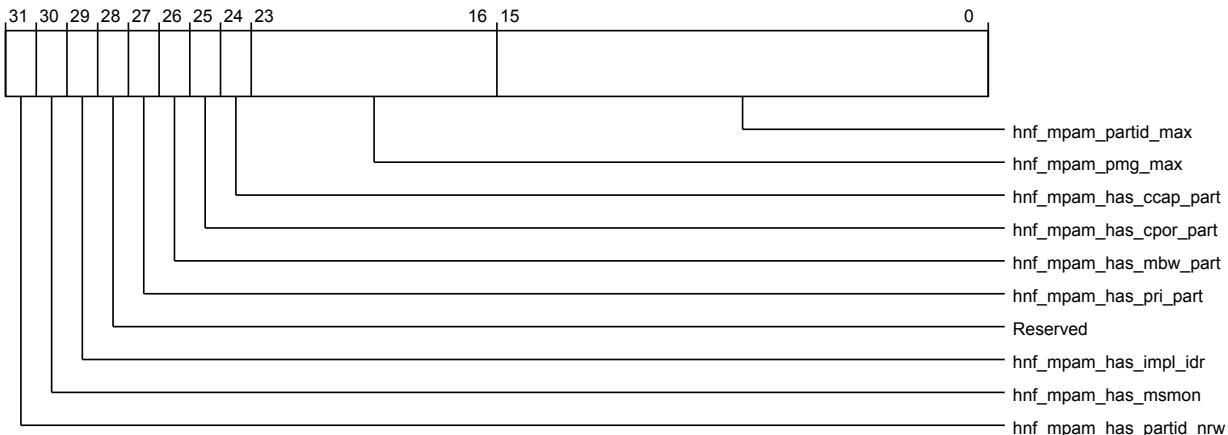


Figure 3-958 por_hnf_mpam_ns_por_hnf_mpam_idr (low)

The following table shows the por_hnf_mpam_idr lower register bit assignments.

Table 3-978 por_hnf_mpam_ns_por_hnf_mpam_idr (low)

Bits	Field name	Description	Type	Reset
31	hnf_mpam_has_partid_nrw	0: HN-F does not support MPAM PARTID Narrowing 1: HN-F supports MPAM PARTID Narrowing	RO	Configuration dependent
30	hnf_mpam_has_msmon	0: MPAM performance monitoring is not supported 1: MPAM performance monitoring is supported	RO	Configuration dependent
29	hnf_mpam_has_impl_idr	0: MPAM implementation specific partitioning features not supported 1: MPAM implementation specific partitioning features supported	RO	Configuration dependent
28	Reserved	Reserved	RO	-
27	hnf_mpam_has_pri_part	0: MPAM priority partitioning is not supported 1: MPAM priority partitioning is supported	RO	Configuration dependent
26	hnf_mpam_has_mbw_part	0: MPAM memory bandwidth partitioning is not supported 1: MPAM memory bandwidth partitioning is supported	RO	Configuration dependent
25	hnf_mpam_has_cpor_part	0: MPAM cache portion partitioning is not supported 1: MPAM cache portion partitioning is supported	RO	Configuration dependent

Table 3-978 por_hnf_mpam_ns_por_hnf_mpam_idr (low) (continued)

Bits	Field name	Description	Type	Reset
24	hnf_mpam_has_ccap_part	0: MPAM cache maximum capacity partitioning is not supported 1: MPAM cache maximum capacity partitioning is supported	RO	Configuration dependent
23:16	hnf_mpam_pmg_max	Maximum value of non-secure PMG supported by this HN-F	RO	Configuration dependent
15:0	hnf_mpam_partid_max	Maximum value of non-secure PARTID supported by this HN-F	RO	Configuration dependent

por_hnf_mpam_iidr

MPAM Implementation ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1018

Register reset 64'b000000000000000000000000100111011

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-959 por_hnf_mpam_ns_por_hnf_mpam_iidr (high)

The following table shows the por_hnf_mpam_iidr higher register bit assignments.

Table 3-979 por_hnf_mpam_ns_por_hnf_mpam_iidr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

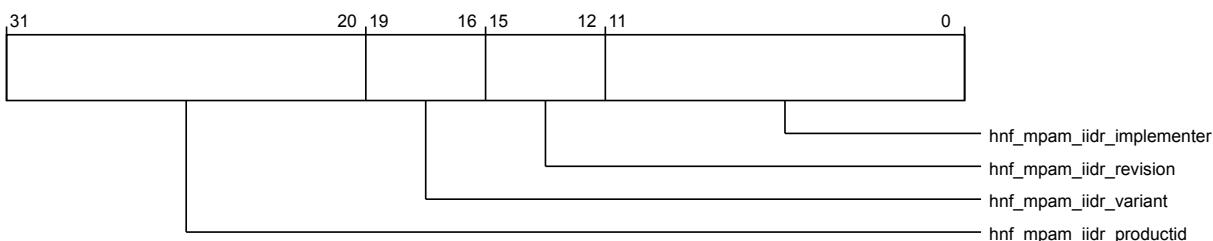


Figure 3-960 por_hnf_mpam_ns_por_hnf_mpam_iidr (low)

The following table shows the por_hnf_mpam_iidr lower register bit assignments.

Table 3-980 por_hnf_mpam_ns_por_hnf_mpam_iidr (low)

Bits	Field name	Description	Type	Reset
31:20	hnf_mpam_iidr_productid	Implementation defined value identifying MPAM memory system component	RO	12'h000
19:16	hnf_mpam_iidr_variant	Implementation defined value identifying major revision of the product	RO	4'b0000
15:12	hnf_mpam_iidr_revision	Implementation defined value identifying minor revision of the product	RO	4'b0000
11:0	hnf_mpam_iidr_implementer	Implementation defined value identifying company that implemented MPAM memory system component	RO	12'h43B

por_hnf_mpam_aidr

MPAM architecture ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1020

Register reset 64'b00010000

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

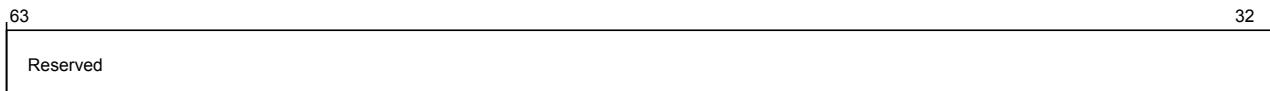


Figure 3-961 por_hnf_mpam_ns_por_hnf_mpam_aidr (high)

The following table shows the por_hnf_mpam_aidr higher register bit assignments.

Table 3-981 por_hnf_mpam_ns_por_hnf_mpam_aidr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

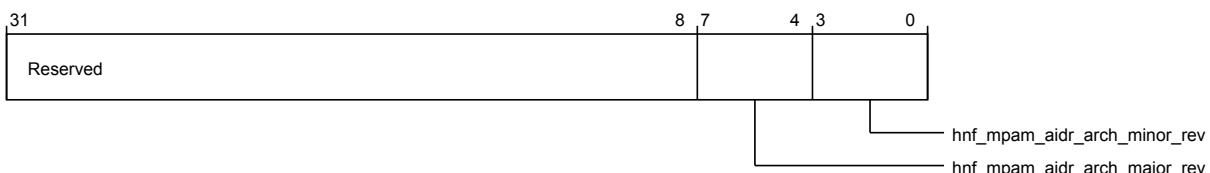


Figure 3-962 por_hnf_mpam_ns_por_hnf_mpam_aidr (low)

The following table shows the por_hnf_mpam_aidr lower register bit assignments.

Table 3-982 por_hnf_mpam_ns_por_hnf_mpam_aistr (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:4	hnf_mpam_aistr_arch_major_rev	Major revision of the MPAM architecture that this memory system component implements	RO	4'b0001
3:0	hnf_mpam_aistr_arch_minor_rev	Minor revision of the MPAM architecture that this memory system component implements	RO	4'b0000

por_hnf_mpam_impl_idr

MPAM Implementation defined partitioning feature ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1028

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

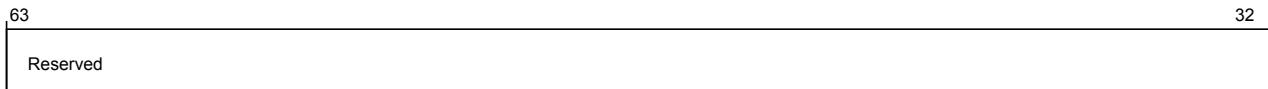


Figure 3-963 por_hnf_mpam_ns_por_hnf_mpam_impl_idr (high)

The following table shows the por_hnf_mpam_impl_idr higher register bit assignments.

Table 3-983 por_hnf_mpam_ns_por_hnf_mpam_impl_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

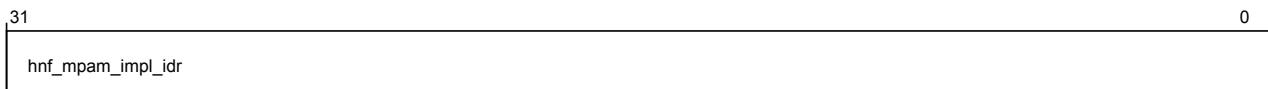


Figure 3-964 por_hnf_mpam_ns_por_hnf_mpam_impl_idr (low)

The following table shows the por_hnf_mpam_impl_idr lower register bit assignments.

Table 3-984 por_hnf_mpam_ns_por_hnf_mpam_impl_idr (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_mpam_impl_idr	Implementation defined partitioning features.	RO	32'h00000000

por_hnf_mpam_cpor_idr

MPAM cache portion partitioning ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1030

Register reset Configuration dependent

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

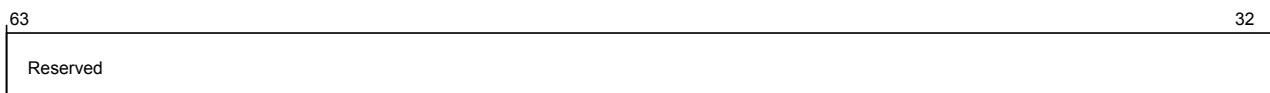


Figure 3-965 por_hnf_mpam_ns_por_hnf_mpam_cpor_idr (high)

The following table shows the por_hnf_mpam_cpor_idr higher register bit assignments.

Table 3-985 por_hnf_mpam_ns_por_hnf_mpam_cpor_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

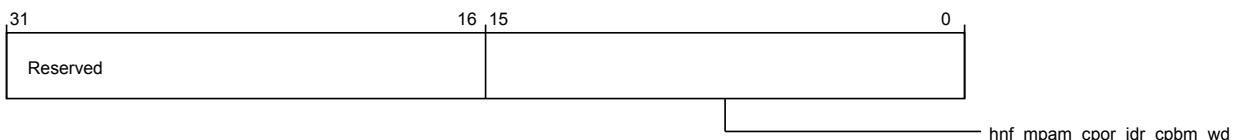


Figure 3-966 por_hnf_mpam_ns_por_hnf_mpam_cpor_idr (low)

The following table shows the por_hnf_mpam_cpor_idr lower register bit assignments.

Table 3-986 por_hnf_mpam_ns_por_hnf_mpam_cpor_idr (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	hnf_mpam_cpor_idr_cpmb_wd	Number of bits in the cache portion partitioning bit map of this device.	RO	Configuration dependent

por_hnf_mpam_ccap_idr

MPAM cache capacity partitioning ID register. This is a shared register for S and NS

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h1038
Register reset	Configuration dependent
Usage constraints	This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

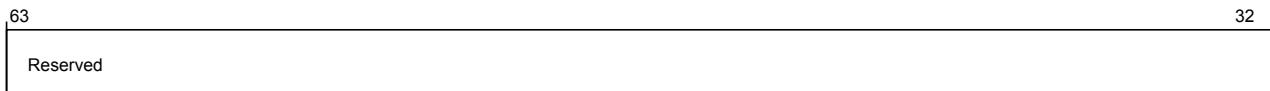


Figure 3-967 por_hnf_mpam_ns_por_hnf_mpam_ccap_idr (high)

The following table shows the por_hnf_mpam_ccap_idr higher register bit assignments.

Table 3-987 por_hnf_mpam_ns_por_hnf_mpam_ccap_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

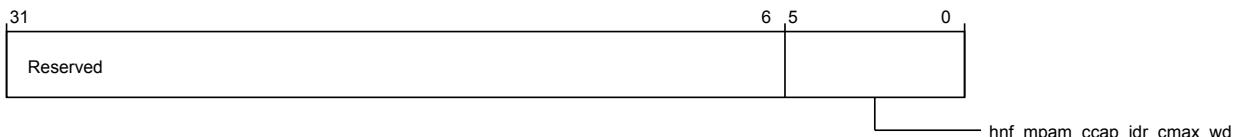


Figure 3-968 por_hnf_mpam_ns_por_hnf_mpam_ccap_idr (low)

The following table shows the por_hnf_mpam_ccap_idr lower register bit assignments.

Table 3-988 por_hnf_mpam_ns_por_hnf_mpam_ccap_idr (low)

Bits	Field name	Description	Type	Reset
31:6	Reserved	Reserved	RO	-
5:0	hnf_mpam_ccap_idr_cmax_wd	Number of fractional bits implemented in the cache capacity partitioning.	RO	Configuration dependent

por_hnf_mpam_mbw_idr

MPAM Memory Bandwidth partitioning ID register. This is a shared register for S and NS

Its characteristics are:

Type	RO
-------------	----

Register width (Bits) 64

Address offset 16'h1040

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-969 por_hnf_mpam_ns_por_hnf_mpam_mbw_idr (high)

The following table shows the por_hnf_mpam_mbw_idr higher register bit assignments.

Table 3-989 por_hnf_mpam_ns_por_hnf_mpam_mbw_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

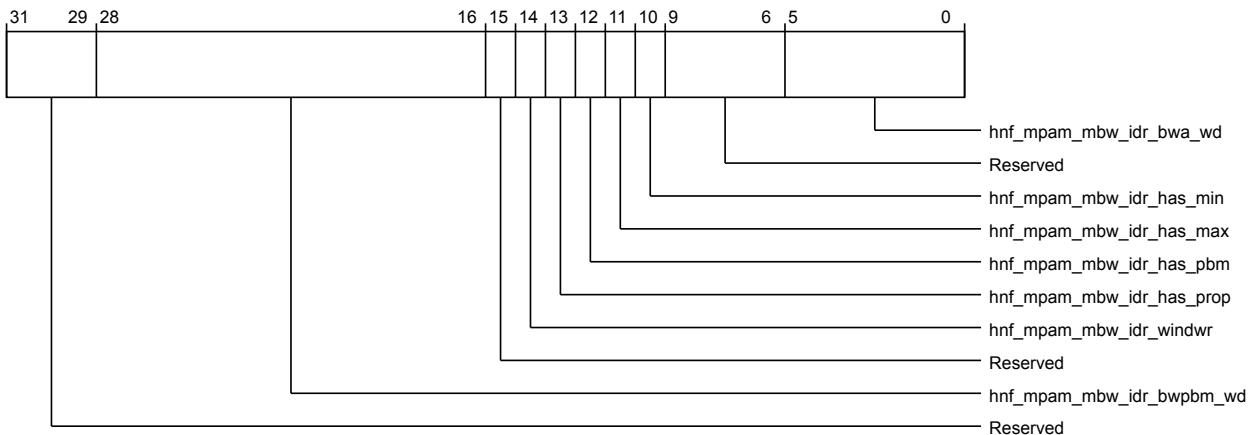


Figure 3-970 por_hnf_mpam_ns_por_hnf_mpam_mbw_idr (low)

The following table shows the por_hnf_mpam_mbw_idr lower register bit assignments.

Table 3-990 por_hnf_mpam_ns_por_hnf_mpam_mbw_idr (low)

Bits	Field name	Description	Type	Reset
31:29	Reserved	Reserved	RO	-
28:16	hnf_mpam_mbw_idr_bwpbm_wd	Number of bits indication portions in MPAMCFG_MBW_PBM register.	RO	13'h0
15	Reserved	Reserved	RO	-

Table 3-990 por_hnf_mpam_ns_por_hnf_mpam_mbw_idr (low) (continued)

Bits	Field name	Description	Type	Reset
14	hnf_mpam_mbw_idr_windwr	0: The bandwidth accounting period should be read from MPAMCFG_MBW_WINDWR register, which might be fixed. 1: The bandwidth accounting width is readable and writable per partition in MPAMCFG_MBW_WINDWR register.	RO	1'h0
13	hnf_mpam_mbw_idr_has_prop	0: There is no memory bandwidth proportional stride control and no MPAMCFG_MBW_PROP register 1: MPAMCFG_MBW_PROP register exists and memory bandwidth proportional stride memory bandwidth allocation scheme is supported.	RO	1'h0
12	hnf_mpam_mbw_idr_has_pbm	0: There is no memory bandwidth portion control and no MPAMCFG_MBW_PBM register 1: MPAMCFG_MBW_PBM register exists and memory bandwidth portion allocation scheme is supported.	RO	1'h0
11	hnf_mpam_mbw_idr_has_max	0: There is no maximum memory bandwidth control and no MPAMCFG_MBW_MAX register 1: MPAMCFG_MBW_MAX register exists and maximum memory bandwidth allocation scheme is supported.	RO	1'h0
10	hnf_mpam_mbw_idr_has_min	0: There is no minimum memory bandwidth control and no MPAMCFG_MBW_MIN register 1: MPAMCFG_MBW_MIN register exists and minimum memory bandwidth allocation scheme is supported.	RO	1'h0
9:6	Reserved	Reserved	RO	-
5:0	hnf_mpam_mbw_idr_bwa_wd	Number of implemented bits in bandwidth allocation fields: MIN, MAX, and STRIDE. Value must be between 1 to 16	RO	4'b0000

por_hnf_mpam_pri_idr

MPAM Priority partitioning ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1048

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-971 por_hnf_mpam_ns_por_hnf_mpam_pri_idr (high)

The following table shows the por_hnf_mpam_pri_idr higher register bit assignments.

Table 3-991 por_hnf_mpam_ns_por_hnf_mpam_pri_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

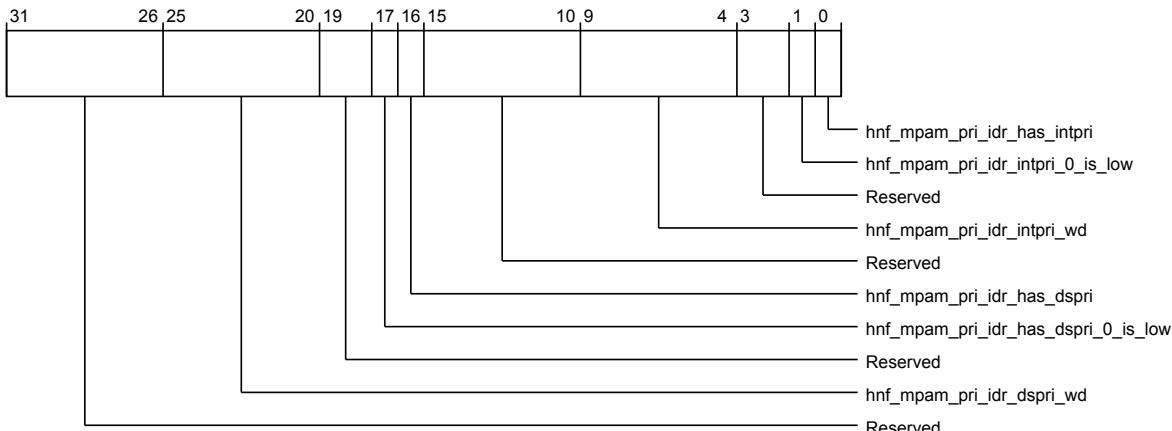


Figure 3-972 por_hnf_mpam_ns_por_hnf_mpam_pri_idr (low)

The following table shows the por_hnf_mpam_pri_idr lower register bit assignments.

Table 3-992 por_hnf_mpam_ns_por_hnf_mpam_pri_idr (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:20	hnf_mpam_pri_idr_dspri_wd	Number of bits in downstream priority field (DSPRI) in MPAMCFG_PRI.	RO	6'h0
19:18	Reserved	Reserved	RO	-
17	hnf_mpam_pri_idr_has_dspri_0_is_low	0: In the DSPRI field, a value of 0 means highest priority. 1: In the DSPRI field, a value of 0 means lowest priority.	RO	1'h0
16	hnf_mpam_pri_idr_has_dspri	0: This memory system component supports priority, but doesn't have a downstream priority (DSPRI) field in MPAMCFG_PRI. 1: This memory system component supports downstream priority and has an DSPRI field.	RO	1'h0
15:10	Reserved	Reserved	RO	-
9:4	hnf_mpam_pri_idr_intpri_wd	Number of bits in the internal priority field (INTPRI) in MPAMCFG_PRI.	RO	6'h0
3:2	Reserved	Reserved	RO	-

Table 3-992 por_hnf_mpam_ns_por_hnf_mpam_pri_idr (low) (continued)

Bits	Field name	Description	Type	Reset
1	hnf_mpam_pri_idr_intpri_0_is_low	0: In the INTPRI field, a value of 0 means highest priority. 1: In the INTPRI field, a value of 0 means lowest priority.	RO	1'h0
0	hnf_mpam_pri_idr_has_intpri	0: This memory system component supports priority, but doesn't have an internal priority field in MPAMCFG_PRI. 1: This memory system component supports internal priority and has an INTPRI field.	RO	1'h0

por_hnf_mpam_partid_nrw_idr

MPAM PARTID narrowing ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1050

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-973 por_hnf_mpam_ns_por_hnf_mpam_partid_nrw_idr (high)

The following table shows the por_hnf_mpam_partid_nrw_idr higher register bit assignments.

Table 3-993 por_hnf_mpam_ns_por_hnf_mpam_partid_nrw_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

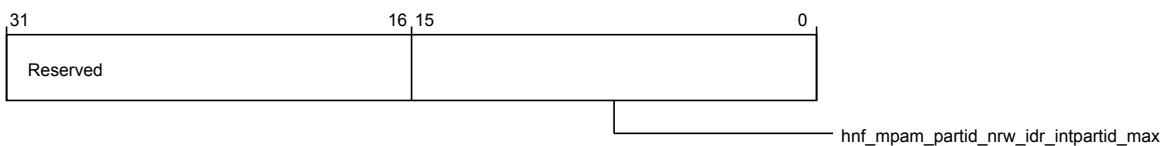


Figure 3-974 por_hnf_mpam_ns_por_hnf_mpam_partid_nrw_idr (low)

The following table shows the por_hnf_mpam_partid_nrw_idr lower register bit assignments.

Table 3-994 por_hnf_mpam_ns_por_hnf_mpam_partid_nrw_idr (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	hnf_mpam_partid_nrw_idr_intpartid_max	This field indicates the largest intPARTID supported in this component.	RO	16'h00

por_hnf_mpam_msmon_idr

MPAM performance monitoring ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1080

Register reset Configuration dependent

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-975 por_hnf_mpam_ns_por_hnf_mpam_msmon_idr (high)

The following table shows the por_hnf_mpam_msmon_idr higher register bit assignments.

Table 3-995 por_hnf_mpam_ns_por_hnf_mpam_msmon_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

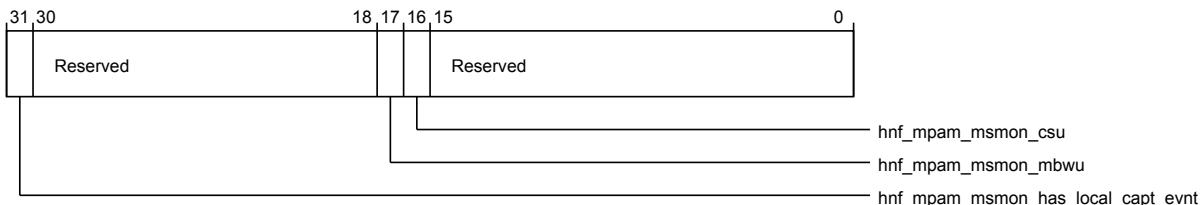


Figure 3-976 por_hnf_mpam_ns_por_hnf_mpam_msmon_idr (low)

The following table shows the por_hnf_mpam_msmon_idr lower register bit assignments.

Table 3-996 por_hnf_mpam_ns_por_hnf_mpam_msmon_idr (low)

Bits	Field name	Description	Type	Reset
31	hnf_mpam_msmon_has_local_capt_evnt	Has the local capture event generator and the MSMON_CAPT_EVNT register.	RO	1'h1
30:18	Reserved	Reserved	RO	-
17	hnf_mpam_msmon_mbwu	This component has a performance monitor for Memory Bandwidth Usage by PARTID and PMG.	RO	Configuration dependent
16	hnf_mpam_msmon_csu	This component has a performance monitor for Cache Storage Usage by PARTID and PMG.	RO	Configuration dependent
15:0	Reserved	Reserved	RO	-

por_hnf_mpam_csumon_idr

MPAM cache storage usage monitor ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1088

Register reset Configuration dependent

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-977 por_hnf_mpam_ns_por_hnf_mpam_csumon_idr (high)

The following table shows the por_hnf_mpam_csumon_idr higher register bit assignments.

Table 3-997 por_hnf_mpam_ns_por_hnf_mpam_csumon_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

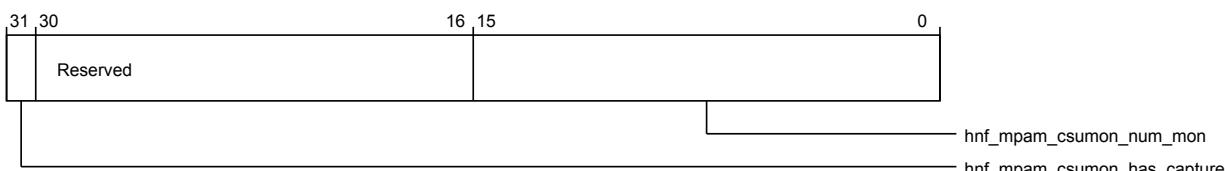


Figure 3-978 por_hnf_mpam_ns_por_hnf_mpam_csumon_idr (low)

The following table shows the por_hnf_mpam_csumon_idr lower register bit assignments.

Table 3-998 por_hnf_mpam_ns_por_hnf_mpam_csumon_idr (low)

Bits	Field name	Description	Type	Reset
31	hnf_mpam_csumon_has_capture	0: MSMON_CSU_CAPTURE is not implemented and there is no support for capture events in this component's CSU monitor feature. 1: This component's CSU monitor feature has an MSMON_CSU_CAPTURE register for every MSMON_CSU and supports the capture event behaviour.	RO	1'h1
30:16	Reserved	Reserved	RO	-
15:0	hnf_mpam_csumon_num_mon	The number of CSU monitoring counters implemented in this component.	RO	Configuration dependent

por_hnf_mpam_mbwumon_idr

MPAM memory bandwidth usage monitor ID register. This is a shared register for S and NS

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h1090

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-979 por_hnf_mpam_ns_por_hnf_mpam_mbwumon_idr (high)

The following table shows the por_hnf_mpam_mbwumon_idr higher register bit assignments.

Table 3-999 por_hnf_mpam_ns_por_hnf_mpam_mbwumon_idr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

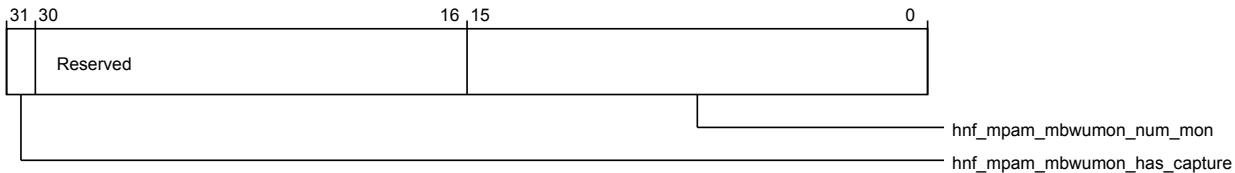


Figure 3-980 por_hnf_mpam_ns_por_hnf_mpam_mbwumon_idr (low)

The following table shows the por_hnf_mpam_mbwumon_idr lower register bit assignments.

Table 3-1000 por_hnf_mpam_ns_por_hnf_mpam_mbwumon_idr (low)

Bits	Field name	Description	Type	Reset
31	hnf_mpam_mbwumon_has_capture	0: MSMON_MBWU_CAPTURE is not implemented and there is no support for capture events in this component's MBWU monitor feature. 1: This component's MBWU monitor feature has an MSMON_MBWU_CAPTURE register for every MSMON_MBWU and supports the capture event behaviour.	RO	1'h0
30:16	Reserved	Reserved	RO	-
15:0	hnf_mpam_mbwumon_num_mon	The number of MBWU monitoring counters implemented in this component.	RO	16'h0

por_hnf_ns_mpam_ecr

MPAM Error Control Register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h10F0

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

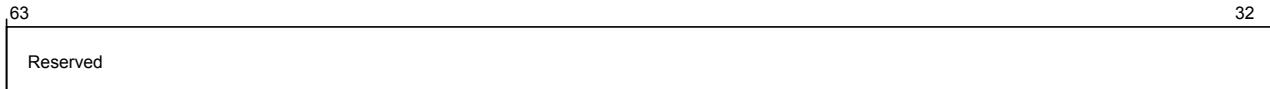


Figure 3-981 por_hnf_mpam_ns_por_hnf_ns_mpam_ecr (high)

The following table shows the por_hnf_ns_mpam_ecr higher register bit assignments.

Table 3-1001 por_hnf_mpam_ns_por_hnf_ns_mpam_ecr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

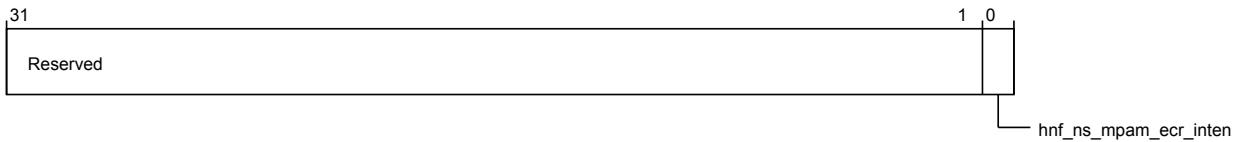


Figure 3-982 por_hnf_mpam_ns_por_hnf_ns_mpam_ecr (low)

The following table shows the por_hnf_ns_mpam_ecr lower register bit assignments.

Table 3-1002 por_hnf_mpam_ns_por_hnf_ns_mpam_ecr (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	hnf_ns_mpam_ecr_inten	Interrupt Enable. When INTEN = 0, MPAM error interrupts are not generated. When INTEN = 1, MPAM error interrupts are generated.	RW	1'h0

por_hnf_ns_mpam_esr

MPAM Error Status Register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h10F8

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

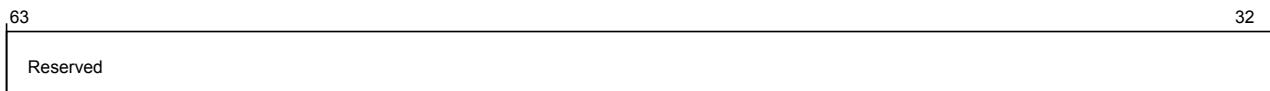


Figure 3-983 por_hnf_mpam_ns_por_hnf_ns_mpam_esr (high)

The following table shows the por_hnf_ns_mpam_esr higher register bit assignments.

Table 3-1003 por_hnf_mpam_ns_por_hnf_ns_mpam_esr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

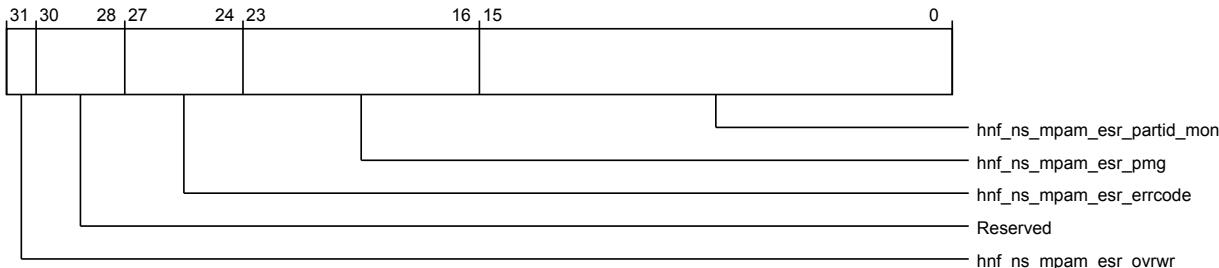


Figure 3-984 por_hnf_ns_mpam_ns_por_hnf_ns_mpam_esr (low)

The following table shows the por_hnf_ns_mpam_esr lower register bit assignments.

Table 3-1004 por_hnf_mpam_ns_por_hnf_ns_mpam_esr (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_mpam_esr_ovrwr	Overwritten. If 0 and ERRCODE is zero, no errors have occurred. If 0 and ERRCODE is non-zero, a single error has occurred and is recorded in this register. If 1 and ERRCODE is non-zero, multiple errors have occurred and this register records the most recent error. The state where this bit is 1 and ERRCODE is zero is not produced by hardware and is only reached when software writes this combination into this register.	RW	1'h0
30:28	Reserved	Reserved	RO	-
27:24	hnf_ns_mpam_esr_errcode	Error code	RW	4'h0
23:16	hnf_ns_mpam_esr_pmg	PMG captured if the error code captures PMG, otherwise 0x0000.	RW	8'h0
15:0	hnf_ns_mpam_esr_partid_mon	PARTID captured if the error code captures PARTID. MON selector captured if the error code captures MON. Otherwise 0x0000.	RW	16'h0

por_hnf_ns_mpamcfg_part_sel

MPAM partition configuration selection register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1100

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

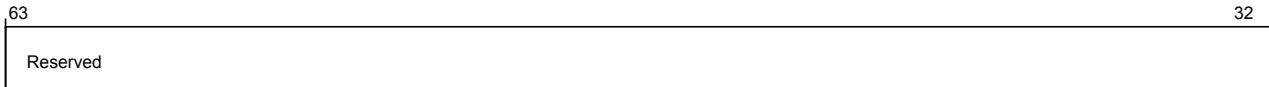


Figure 3-985 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_part_sel (high)

The following table shows the por_hnf_ns_mpamcfg_part_sel higher register bit assignments.

Table 3-1005 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_part_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

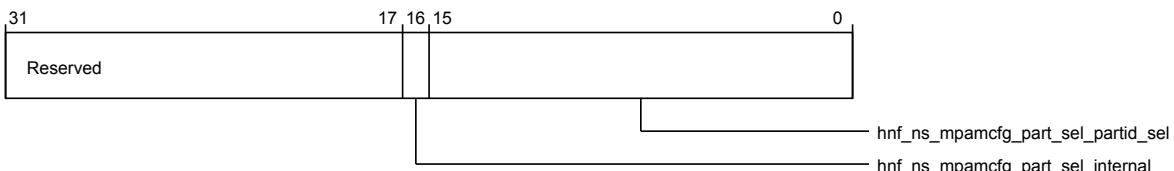


Figure 3-986 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_part_sel (low)

The following table shows the por_hnf_ns_mpamcfg_part_sel lower register bit assignments.

Table 3-1006 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_part_sel (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	hf_ns_mpamcfg_part_sel_internal	If MPAMF_IDR.HAS_PARTID_NRW = 0, this field is RAZ/WI. If MPAMF_IDR.HAS_PARTID_NRW = 1, this bit decides how to interpret PARTID_SEL.	RW	1'h0
15:0	hf_ns_mpamcfg_part_sel_partid_sel	Selects the partition ID to configure.	RW	16'h0

por_hnf_ns_mpamcfg_cmax

MPAM cache maximum capacity partition configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1108

Register reset 64'b111111

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

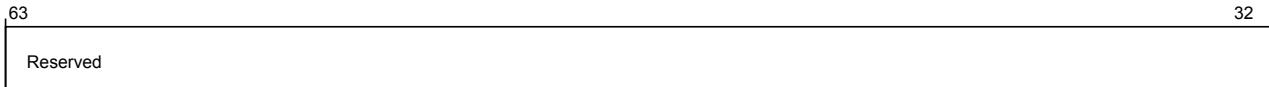


Figure 3-987 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cmax (high)

The following table shows the por_hnf_ns_mpamcfg_cmax higher register bit assignments.

Table 3-1007 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cmax (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

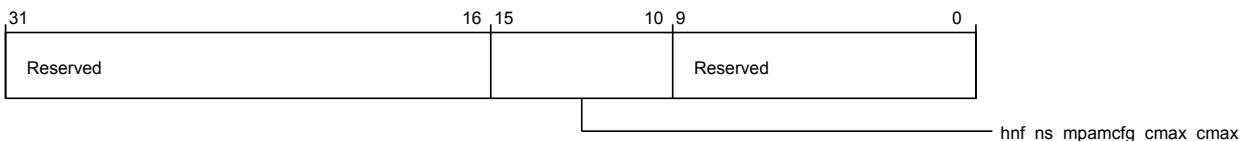


Figure 3-988 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cmax (low)

The following table shows the por_hnf_ns_mpamcfg_cmax lower register bit assignments.

Table 3-1008 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cmax (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:10	hnf_ns_mpamcfg_cmax_cmax	Maximum cache capacity usage in fixed-point fraction of the cache capacity by the partition selected by MPAMCFG_PART_SEL.	RW	6'h3f
9:0	Reserved	Reserved	RO	-

por_hnf_ns_mpamcfg_mbw_min

MPAM memory minimum bandwidth partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1200

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

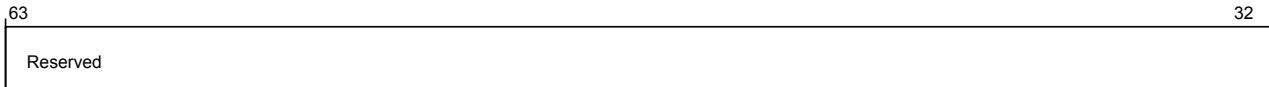


Figure 3-989 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_min (high)

The following table shows the por_hnf_ns_mpamcfg_mbw_min higher register bit assignments.

Table 3-1009 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_min (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

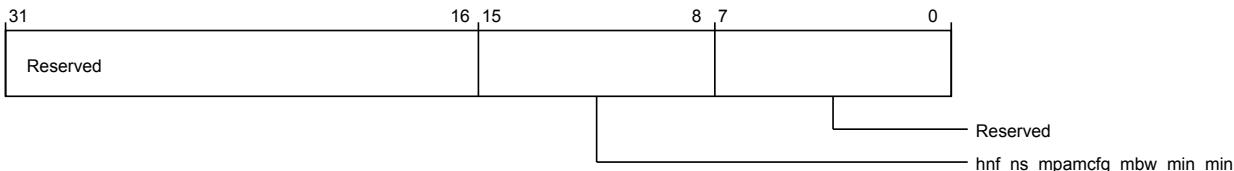


Figure 3-990 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_min (low)

The following table shows the por_hnf_ns_mpamcfg_mbw_min lower register bit assignments.

Table 3-1010 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_min (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:8	hnf_ns_mpamcfg_mbw_min_min	Memory minimum bandwidth allocated to the partition selected by MPAMCFG_PART_SEL.	RW	8'h0
7:0	Reserved	Reserved	RO	-

por_hnf_ns_mpamcfg_mbw_max

MPAM memory maximum bandwidth partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1208

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-991 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_max (high)

The following table shows the por_hnf_ns_mpamcfg_mbw_max higher register bit assignments.

Table 3-1011 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_max (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

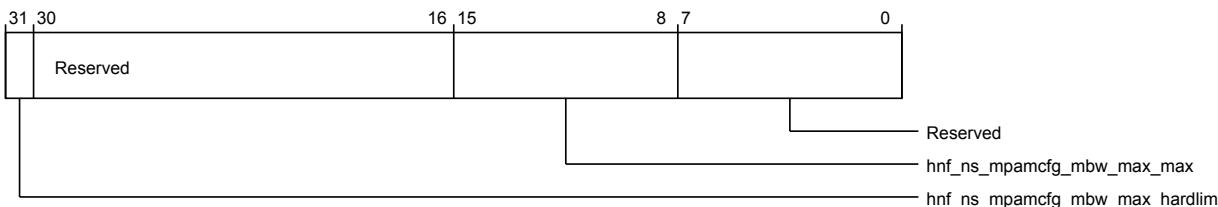


Figure 3-992 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_max (low)

The following table shows the por_hnf_ns_mpamcfg_mbw_max lower register bit assignments.

Table 3-1012 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_max (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_mpamcfg_mbw_max_hardlim	0: When MAX bandwidth is exceeded, the partition may contend with a low preference for downstream bandwidth beyond its maximum bandwidth. 1: When MAX bandwidth is exceeded, the partition may not be use any more bandwidth until its memory bandwidth measurement falls below the maximum limit.	RW	1'h0
30:16	Reserved	Reserved	RO	-
15:8	hnf_ns_mpamcfg_mbw_max_max	Memory maximum bandwidth allocated to the partition selected by MPAMCFG_PART_SEL.	RW	8'h0
7:0	Reserved	Reserved	RO	-

por_hnf_ns_mpamcfg_mbw_winwd

MPAM memory bandwidth partitioning window width register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset	16'h1220
Register reset	64'b0
Usage constraints	This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

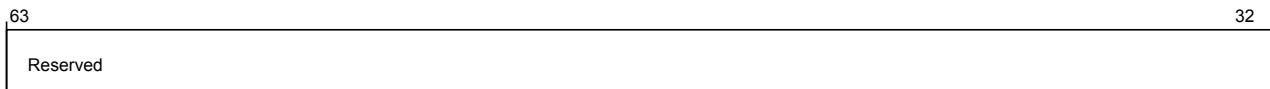


Figure 3-993 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_winwd (high)

The following table shows the por_hnf_ns_mpamcfg_mbw_winwd higher register bit assignments.

Table 3-1013 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_winwd (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

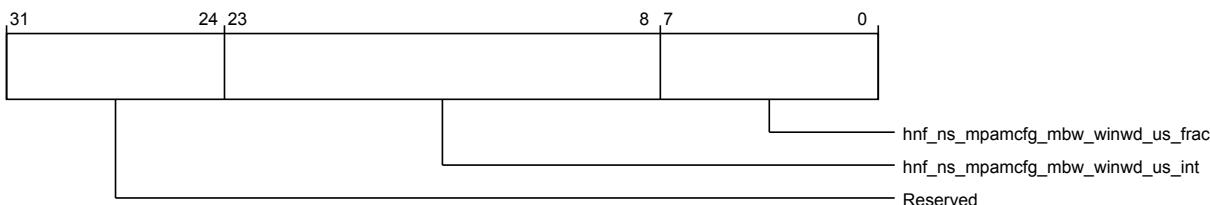


Figure 3-994 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_winwd (low)

The following table shows the por_hnf_ns_mpamcfg_mbw_winwd lower register bit assignments.

Table 3-1014 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_winwd (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:8	hnf_ns_mpamcfg_mbw_winwd_us_int	Memory bandwidth accounting period integer microseconds.	RW	16'h0
7:0	hnf_ns_mpamcfg_mbw_winwd_us_frac	Memory bandwidth accounting period fractions of a microsecond.	RW	8'h0

por_hnf_ns_mpamcfg_pri

MPAM priority partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1400
Register reset	64'b0
Usage constraints	This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

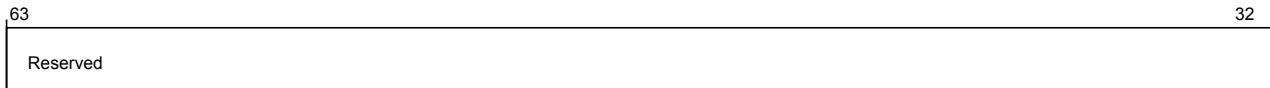


Figure 3-995 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_pri (high)

The following table shows the por_hnf_ns_mpamcfg_pri higher register bit assignments.

Table 3-1015 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_pri (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

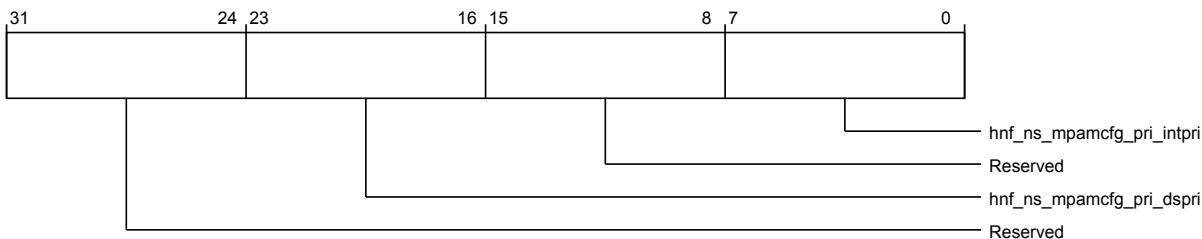


Figure 3-996 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_pri (low)

The following table shows the por_hnf_ns_mpamcfg_pri lower register bit assignments.

Table 3-1016 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_pri (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_ns_mpamcfg_pri_dspri	If HAS_DSPRI is 1, this field is a priority value applied to downstream communications from this memory system component for transactions of the partition selected by MPAMCFG_PART_SEL.	RW	8'h0
15:8	Reserved	Reserved	RO	-
7:0	hnf_ns_mpamcfg_pri_intpri	If HAS_INTPRI is 1, this field is a priority value applied internally inside this memory system component for transactions of the partition selected by MPAMCFG_PART_SEL.	RW	8'h0

por_hnf_ns_mpamcfg_mbw_prop

Memory bandwidth proportional stride partitioning configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1500

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

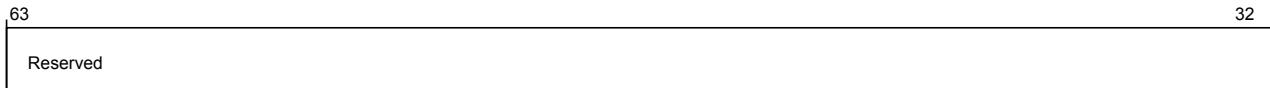


Figure 3-997 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_prop (high)

The following table shows the por_hnf_ns_mpamcfg_mbw_prop higher register bit assignments.

Table 3-1017 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_prop (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

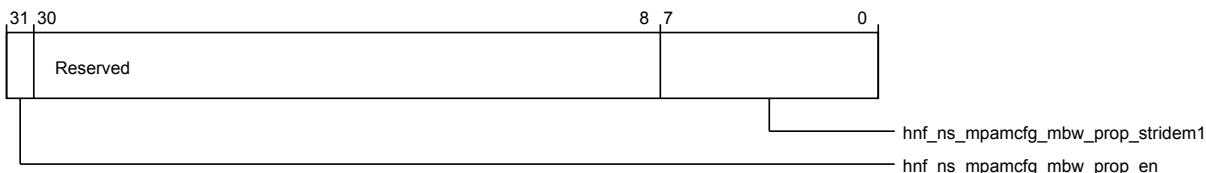


Figure 3-998 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_prop (low)

The following table shows the por_hnf_ns_mpamcfg_mbw_prop lower register bit assignments.

Table 3-1018 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_mbw_prop (low)

Bits	Field name	Description	Type	Reset
31	hf_ns_mpamcfg_mbw_prop_en	0: The selected partition is not regulated by proportional stride bandwidth partitioning. 1: The selected partition has bandwidth usage regulated by proportional stride bandwidth partitioning as controlled by STRIDEM1.	RW	1'h0
30:8	Reserved	Reserved	RO	-
7:0	hf_ns_mpamcfg_mbw_prop_stridem1	Normalized cost of a bandwidth consumption by the partition. STRIDEM1 is the stride for the partition minus one.	RW	8'h0

por_hnf_ns_mpamcfg_intpartid

MPAM internal partition narrowing configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset	16'h1600
Register reset	64'b0
Usage constraints	This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-999 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_intpartid (high)

The following table shows the por_hnf_ns_mpamcfg_intpartid higher register bit assignments.

Table 3-1019 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_intpartid (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

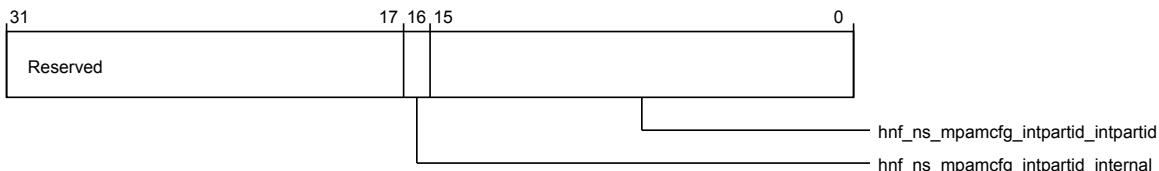


Figure 3-1000 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_intpartid (low)

The following table shows the por_hnf_ns_mpamcfg_intpartid lower register bit assignments.

Table 3-1020 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_intpartid (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	hnf_ns_mpamcfg_intpartid_internal	This bit must be 1 when written to the register. If written as 0, the write will not update the reqPARTID to intPARTID association.	RW	1'h0
15:0	hnf_ns_mpamcfg_intpartid_intpartid	This field contains the intPARTID mapped to the reqPARTID in MPAMCFG_PART_SEL.	RW	16'h0

por_hnf_ns_msmon_cfg_mon_sel

Memory system performance monitor selection register. This register is a banked separately for S and NS. Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1800
Register reset	64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

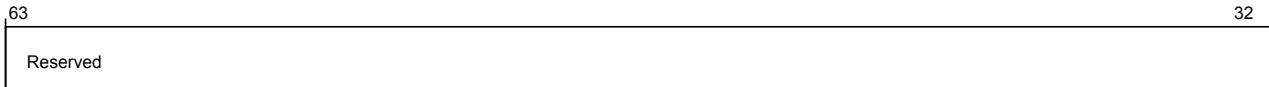


Figure 3-1001 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mon_sel (high)

The following table shows the por_hnf_ns_msmon_cfg_mon_sel higher register bit assignments.

Table 3-1021 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mon_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

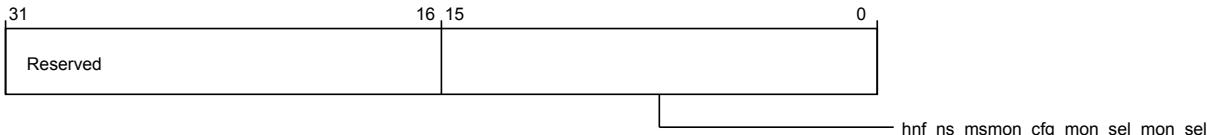


Figure 3-1002 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mon_sel (low)

The following table shows the por_hnf_ns_msmon_cfg_mon_sel lower register bit assignments.

Table 3-1022 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mon_sel (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	hnf_ns_msmon_cfg_mon_sel_mon_sel	Selects the performance monitor to configure.	RW	16'h0

por_hnf_ns_msmon_capt_evtnt

Memory system performance monitoring capture event generation register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1808

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-1003 por_hnf_mpam_ns_por_hnf_ns_msmon_capt_evnt (high)

The following table shows the por_hnf_ns_msmon_capt_evnt higher register bit assignments.

Table 3-1023 por_hnf_mpam_ns_por_hnf_ns_msmon_capt_evnt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

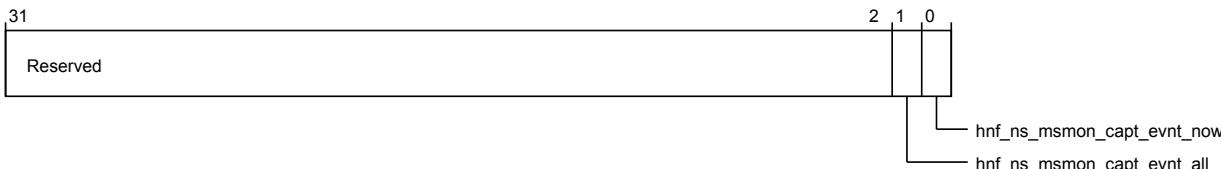


Figure 3-1004 por_hnf_mpam_ns_por_hnf_ns_msmon_capt_evnt (low)

The following table shows the por_hnf_ns_msmon_capt_evnt lower register bit assignments.

Table 3-1024 por_hnf_mpam_ns_por_hnf_ns_msmon_capt_evnt (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1	hnf_ns_msmon_capt_evnt_all	In secure version, if ALL written as 1 and NOW is also written as 1, signal a capture event to secure and non-secure monitors in this memory system component with CAPT_EVNT = 7. If written as 0 and NOW is written as 1, signal a capture event to secure monitors in this memory system component with CAPT_EVNT = 7. In non-secure version if NOW is written as 1, signal a capture event to non-secure monitors in this memory system component with CAPT_EVNT = 7.	RW	1'h0
0	hnf_ns_msmon_capt_evnt_now	When written as 1, this bit causes an event to all monitors in this memory system component with CAPT_EVNT set to the value of 7. When this bit is written as 0, no event is signalled.	RW	1'h0

por_hnf_ns_msmon_cfg_csu_filt

Memory system performance monitor configure cache storage usage monitor filter register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1810

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

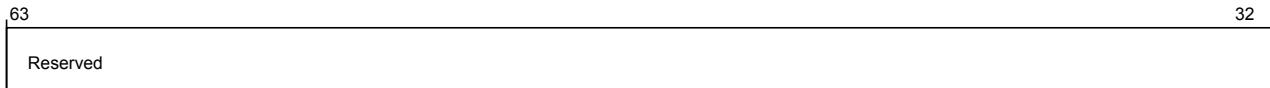


Figure 3-1005 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_flt (high)

The following table shows the por_hnf_ns_msmon_cfg_csu_flt higher register bit assignments.

Table 3-1025 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_flt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

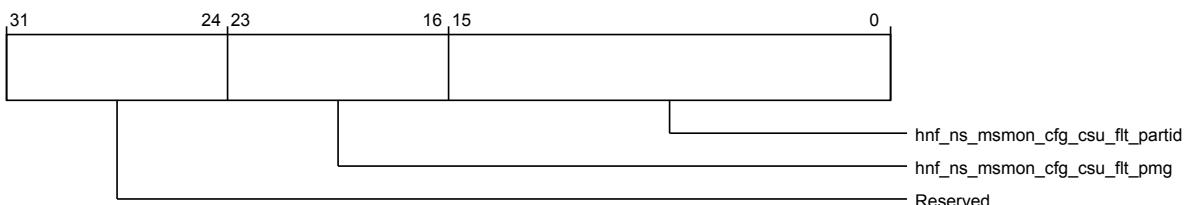


Figure 3-1006 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_flt (low)

The following table shows the por_hnf_ns_msmon_cfg_csu_flt lower register bit assignments.

Table 3-1026 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_flt (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_ns_msmon_cfg_csu_flt_pmg	Configures the cache storage usage performance monitor to a PMG. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures storage usage by cache lines labelled with both the configured PARTID and PMG.	RW	8'h0
15:0	hnf_ns_msmon_cfg_csu_flt_partid	Configures the cache storage usage performance monitor to a PARTID. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures the storage usage by cache lines labelled with both the configured PARTID and PMG.	RW	16'h0

por_hnf_ns_msmon_cfg_csu_ctl

Memory system performance monitor configure cache storage usage monitor control register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1818

Register reset 64'b000000000000000010011

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-1007 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_ctl (high)

The following table shows the por_hnf_ns_msmon_cfg_csu_ctl higher register bit assignments.

Table 3-1027 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

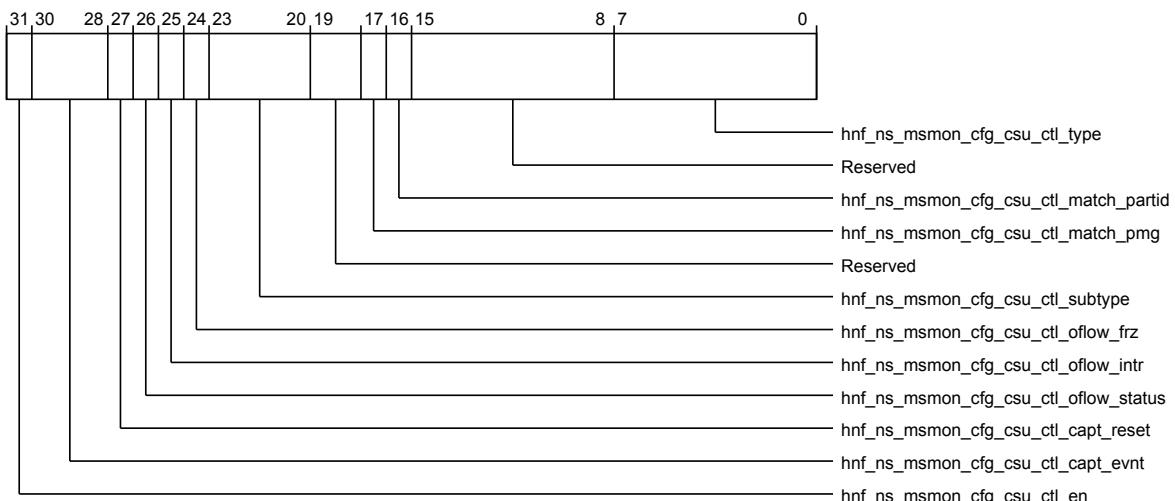


Figure 3-1008 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_ctl (low)

The following table shows the por_hnf_ns_msmon_cfg_csu_ctl lower register bit assignments.

Table 3-1028 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_csu_ctl (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_msmon_cfg_csu_ctl_en	0: The monitor is disabled and must not collect any information. 1: The monitor is enabled to collect information according to its configuration.	RW	1'h0
30:28	hnf_ns_msmon_cfg_csu_ctl_capt_evt	Select the event that triggers capture from the following: 0: No capture event is triggered. 1: External capture event 1 (optional but recommended)	RW	3'h0
27	hnf_ns_msmon_cfg_csu_ctl_capt_reset	Capture is not implemented for the CSU monitor type.	RW	1'h0
26	hnf_ns_msmon_cfg_csu_ctl_oflow_status	0: No overflow has occurred. 1: At least one overflow has occurred since this bit was last written.	RW	1'h0
25	hnf_ns_msmon_cfg_csu_ctl_oflow_intr	0: No interrupt. 1: On overflow, an implementation-specific interrupt is signalled.	RW	1'h0
24	hnf_ns_msmon_cfg_csu_ctl_oflow_frz	0: Monitor count wraps on overflow. 1: Monitor count freezes on overflow. The frozen value may be 0 or another value if the monitor overflowed with an increment larger than 1.	RW	1'h0
23:20	hnf_ns_msmon_cfg_csu_ctl_subtype	Not currently used for CSU monitors, but reserved for future use.	RW	4'h0
19:18	Reserved	Reserved	RO	-
17	hnf_ns_msmon_cfg_csu_ctl_match_pmg	0: Monitor storage used by all PMG values. 1: Only monitor storage used with the PMG value matching MSMON_CFG_CSU_FLT.PMG.	RW	1'h0
16	hnf_ns_msmon_cfg_csu_ctl_match_partid	0: Monitor storage used by all PARTIDs. 1: Only monitor storage used with the PARTID matching MSMON_CFG_CSU_FLT.PARTID.	RW	1'h0
15:8	Reserved	Reserved	RO	-
7:0	hnf_ns_msmon_cfg_csu_ctl_type	Read-only: Constant type indicating the type of the monitor. CSU monitor is TYPE = 0x43.	RW	8'h43

por_hnf_ns_msmon_cfg_mbwu_filt

Memory system performance monitor configure memory bandwidth usage monitor filter register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1820

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

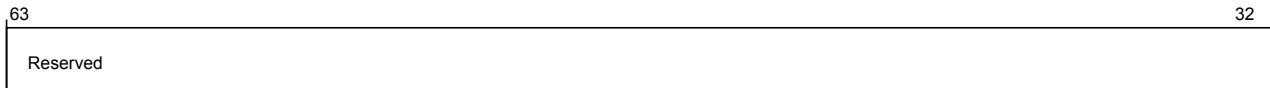


Figure 3-1009 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_flt (high)

The following table shows the por_hnf_ns_msmon_cfg_mbwu_flt higher register bit assignments.

Table 3-1029 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_flt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

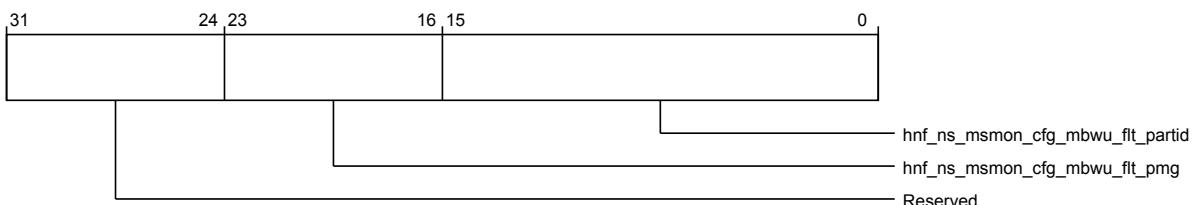


Figure 3-1010 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_flt (low)

The following table shows the por_hnf_ns_msmon_cfg_mbwu_flt lower register bit assignments.

Table 3-1030 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_flt (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_ns_msmon_cfg_mbwu_flt_pmg	Configures the memory bandwidth usage performance monitor to a PMG. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures the memory bandwidth used by requests labelled with both the configured PARTID and PMG.	RW	8'h0
15:0	hnf_ns_msmon_cfg_mbwu_flt_partid	Configures the memory bandwidth usage performance monitor to a PARTID. The monitor selected by MSMON_CFG_MON_SEL.MON_SEL counts or measures the memory bandwidth used by requests labelled with both the configured PARTID and PMG.	RW	16'h0

por_hnf_ns_msmon_cfg_mbwu_ctl

Memory system performance monitor configure memory bandwidth usage monitor control register. This register is a banked separately for S and NS

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1828
Register reset	64'b000000000000000010010
Usage constraints	This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

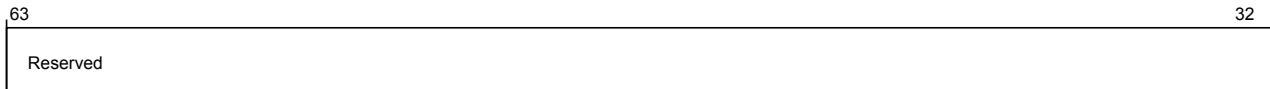


Figure 3-1011 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_ctl (high)

The following table shows the por_hnf_ns_msmon_cfg_mbwu_ctl higher register bit assignments.

Table 3-1031 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

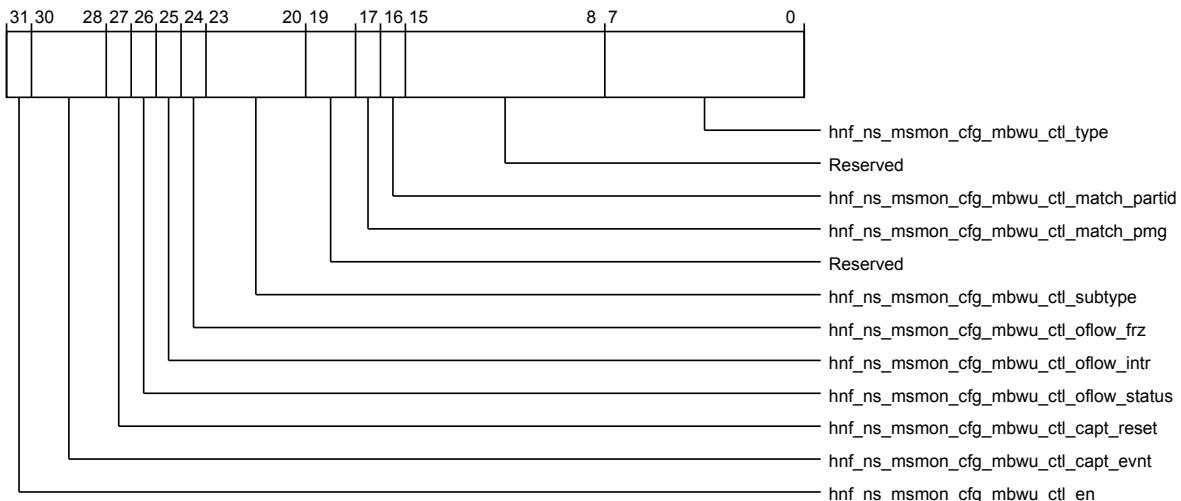


Figure 3-1012 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_ctl (low)

The following table shows the por_hnf_ns_msmon_cfg_mbwu_ctl lower register bit assignments.

Table 3-1032 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_ctl (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_msmon_cfg_mbwu_ctl_en	0: The monitor is disabled and must not collect any information. 1: The monitor is enabled to collect information according to its configuration.	RW	1'h0
30:28	hnf_ns_msmon_cfg_mbwu_ctl_capt_evnt	Select the event that triggers capture from the following: 0: No capture event is triggered. 1: External capture event 1 (optional but recommended)	RW	3'h0
27	hnf_ns_msmon_cfg_mbwu_ctl_capt_reset	0: Monitor is not reset on capture. 1: Monitor is reset on capture.	RW	1'h0
26	hnf_ns_msmon_cfg_mbwu_ctl_oflow_status	0: No overflow has occurred. 1: At least one overflow has occurred since this bit was last written.	RW	1'h0
25	hnf_ns_msmon_cfg_mbwu_ctl_oflow_intr	0: No interrupt. 1: On overflow, an implementation-specific interrupt is signalled.	RW	1'h0
24	hnf_ns_msmon_cfg_mbwu_ctl_oflow_frz	0: Monitor count wraps on overflow. 1: Monitor count freezes on overflow. The frozen value may be 0 or another value if the monitor overflowed with an increment larger than 1.	RW	1'h0
23:20	hnf_ns_msmon_cfg_mbwu_ctl_subtype	A monitor can have other event matching criteria. The meaning of values in this field varies by monitor type. The MBWU monitor type supports: 0: Do not count any bandwidth. 1: Count bandwidth used by memory reads 2: Count bandwidth used by memory writes 3: Count bandwidth used by memory reads and memory writes All other values are reserved and behaviour of a monitor with SUBTYPE set to one of the reserved values is UNPREDICTABLE.	RW	4'h0
19:18	Reserved	Reserved	RO	-
17	hnf_ns_msmon_cfg_mbwu_ctl_match_pmg	0: Monitor bandwidth used by all PMG values. 1: Only monitor bandwidth used with the PMG value matching MSMON_CFG_CSU_FLT.PMG.	RW	1'h0
16	hnf_ns_msmon_cfg_mbwu_ctl_match_partid	0: Monitor bandwidth used by all PARTIDs. 1: Only monitor bandwidth used with the PARTID matching MSMON_CFG_MBWU_FLT.PARTID.	RW	1'h0

Table 3-1032 por_hnf_mpam_ns_por_hnf_ns_msmon_cfg_mbwu_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
15:8	Reserved	Reserved	RO	-
7:0	hnf_ns_msmon_cfg_mbwu_ctl_type	Read-only: Constant type indicating the type of the monitor. MBWU monitor is TYPE = 0x42.	RW	8'h42

por_hnf_ns_msmon_csu

Memory system performance monitor cache storage usage monitor register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1840

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

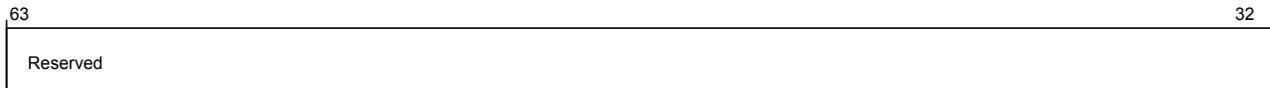


Figure 3-1013 por_hnf_mpam_ns_por_hnf_ns_msmon_csu (high)

The following table shows the por_hnf_ns_msmon_csu higher register bit assignments.

Table 3-1033 por_hnf_mpam_ns_por_hnf_ns_msmon_csu (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

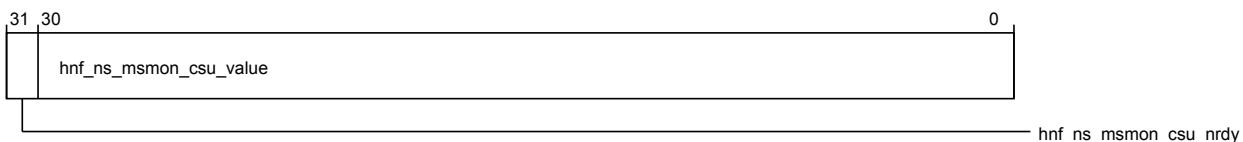


Figure 3-1014 por_hnf_mpam_ns_por_hnf_ns_msmon_csu (low)

The following table shows the por_hnf_ns_msmon_csu lower register bit assignments.

Table 3-1034 por_hnf_mpam_ns_por_hnf_ns_msmon_csu (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_msmon_csu_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_ns_msmon_csu_value	Cache storage usage value if NRDY is 0. Invalid if NRDY is 1. VALUE is the cache storage usage in bytes.	RW	31'h0

por_hnf_ns_msmon_csu_capture

Memory system performance monitor cache storage usage capture register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1848

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

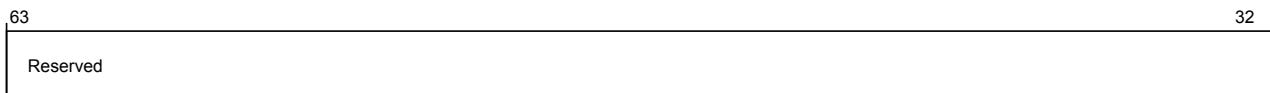


Figure 3-1015 por_hnf_mpam_ns_por_hnf_ns_msmon_csu_capture (high)

The following table shows the por_hnf_ns_msmon_csu_capture higher register bit assignments.

Table 3-1035 por_hnf_mpam_ns_por_hnf_ns_msmon_csu_capture (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

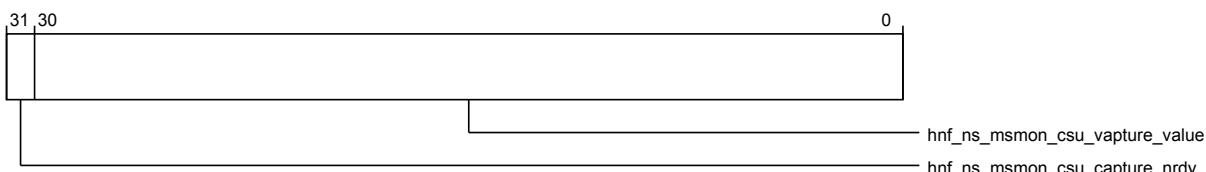


Figure 3-1016 por_hnf_mpam_ns_por_hnf_ns_msmon_csu_capture (low)

The following table shows the por_hnf_ns_msmon_csu_capture lower register bit assignments.

Table 3-1036 por_hnf_mpam_ns_por_hnf_ns_msmon_csu_capture (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_msmon_csu_capture_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_ns_msmon_csu_vapture_value	Cache storage usage value if NRDY is 0. Invalid if NRDY is 1. VALUE is the cache storage usage in bytes.	RW	31'h0

por_hnf_ns_msmon_mbwu

Memory system performance monitor memory bandwidth usage monitor register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1860

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-1017 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu (high)

The following table shows the por_hnf_ns_msmon_mbwu higher register bit assignments.

Table 3-1037 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

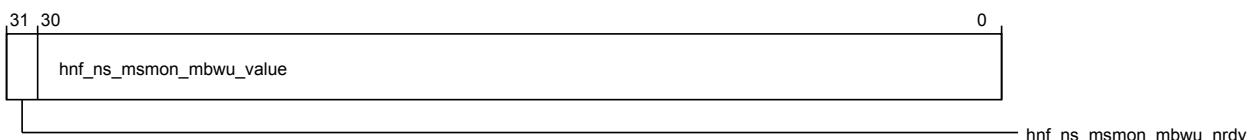


Figure 3-1018 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu (low)

The following table shows the por_hnf_ns_msmon_mbwu lower register bit assignments.

Table 3-1038 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_msmon_mbwu_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_ns_msmon_mbwu_value	Memory channel bandwidth value if NRDY is 0. Invalid if NRDY is 1. VALUE is the memory channel bandwidth usage in megabytes.	RW	31'h0

por_hnf_ns_msmon_mbwu_capture

Memory system performance monitor memory bandwidth usage capture register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1868

Register reset 64'b0

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.



Figure 3-1019 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu_capture (high)

The following table shows the por_hnf_ns_msmon_mbwu_capture higher register bit assignments.

Table 3-1039 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu_capture (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

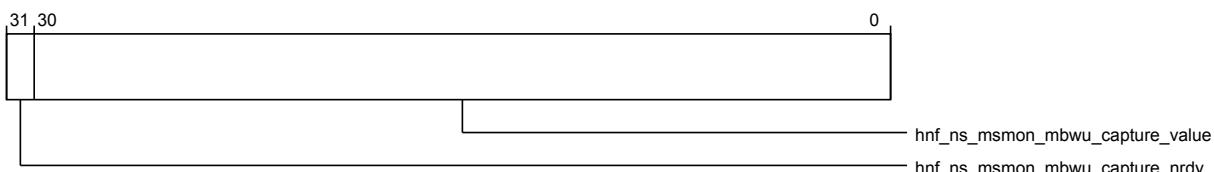


Figure 3-1020 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu_capture (low)

The following table shows the por_hnf_ns_msmon_mbwu_capture lower register bit assignments.

Table 3-1040 por_hnf_mpam_ns_por_hnf_ns_msmon_mbwu_capture (low)

Bits	Field name	Description	Type	Reset
31	hnf_ns_msmon_mbwu_capture_nrdy	Indicates that the monitor does not have accurate data, possibly because insufficient time has elapsed since the monitor was last configured.	RW	1'h0
30:0	hnf_ns_msmon_mbwu_capture_value	Memory channel bandwidth value if NRDY is 0. Invalid if NRDY is 1. VALUE is the memory channel bandwidth usage in megabytes.	RW	31'h0

por_hnf_ns_mpamcfg_cpbm

MPAM cache portion bitmap partition configuration register. This register is a banked separately for S and NS

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b1111111111111111

Usage constraints This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

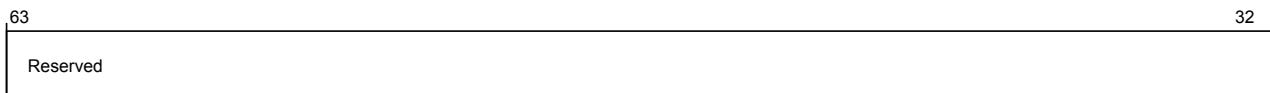


Figure 3-1021 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cpbm (high)

The following table shows the por_hnf_ns_mpamcfg_cpbm higher register bit assignments.

Table 3-1041 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cpbm (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

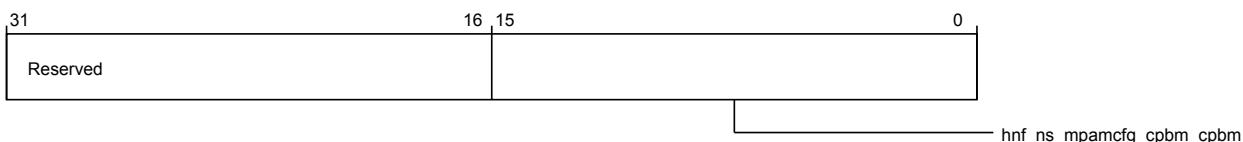


Figure 3-1022 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cpbm (low)

The following table shows the por_hnf_ns_mpamcfg_cpbm lower register bit assignments.

Table 3-1042 por_hnf_mpam_ns_por_hnf_ns_mpamcfg_cpbm (low)

Bits	Field name	Description	Type	Reset
31:16	Reserved	Reserved	RO	-
15:0	hnf_ns_mpamcfg_cpbm_cpbm	Bitmap of portions of cache capacity allocable by the partition selected by MPAMCFG_PART_SEL. NOTE: CPBM can not be all zeros for any PARTID.	RW	16'hFFFF

3.3.17 RN-I register descriptions

This section lists the RN-I registers.

por_rni_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

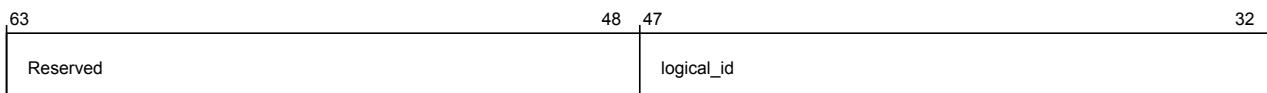


Figure 3-1023 por_rni_por_rni_node_info (high)

The following table shows the por_rni_node_info higher register bit assignments.

Table 3-1043 por_rni_por_rni_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

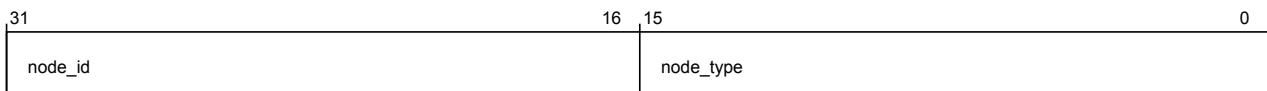


Figure 3-1024 por_rni_por_rni_node_info (low)

The following table shows the por_rni_node_info lower register bit assignments.

Table 3-1044 por_rni_por_rni_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h000A

por_rni_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

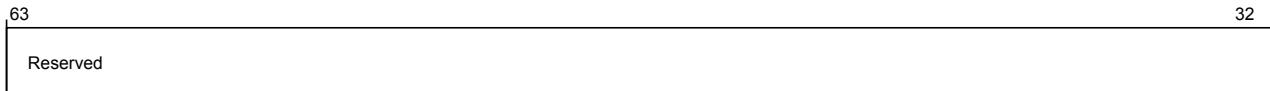


Figure 3-1025 por_rni_por_rni_child_info (high)

The following table shows the por_rni_child_info higher register bit assignments.

Table 3-1045 por_rni_por_rni_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

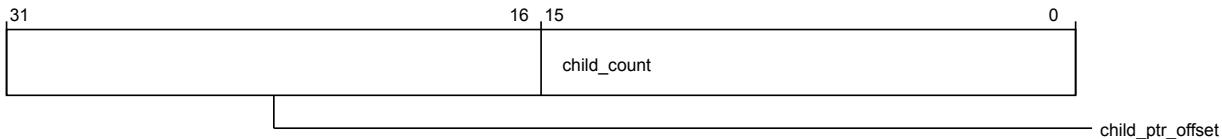


Figure 3-1026 por_rni_por_rni_child_info (low)

The following table shows the por_rni_child_info lower register bit assignments.

Table 3-1046 por_rni_por_rni_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'h0

por_rni_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-1027 por_rni_por_rni_secure_register_groups_override (high)

The following table shows the por_rni_secure_register_groups_override higher register bit assignments.

Table 3-1047 por_rni_por_rni_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

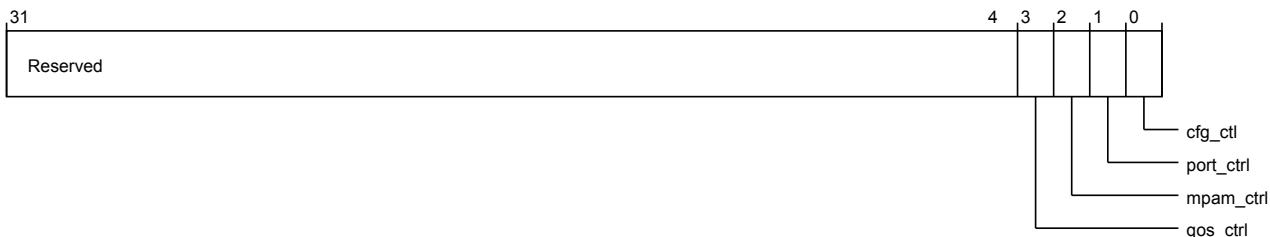


Figure 3-1028 por_rni_por_rni_secure_register_groups_override (low)

The following table shows the por_rni_secure_register_groups_override lower register bit assignments.

Table 3-1048 por_rni_por_rni_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	qos_ctrl	Allows non-secure access to secure QoS control registers	RW	1'b0
2	mpam_ctrl	Allows non-secure access to secure AXI port MPAM override register	RW	1'b0
1	port_ctrl	Allows non-secure access to secure AXI port control registers	RW	1'b0
0	cfg_ctl	Allows non-secure access to secure configuration control register	RW	1'b0

por_rni_unit_info

Provides component identification information for RN-I.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

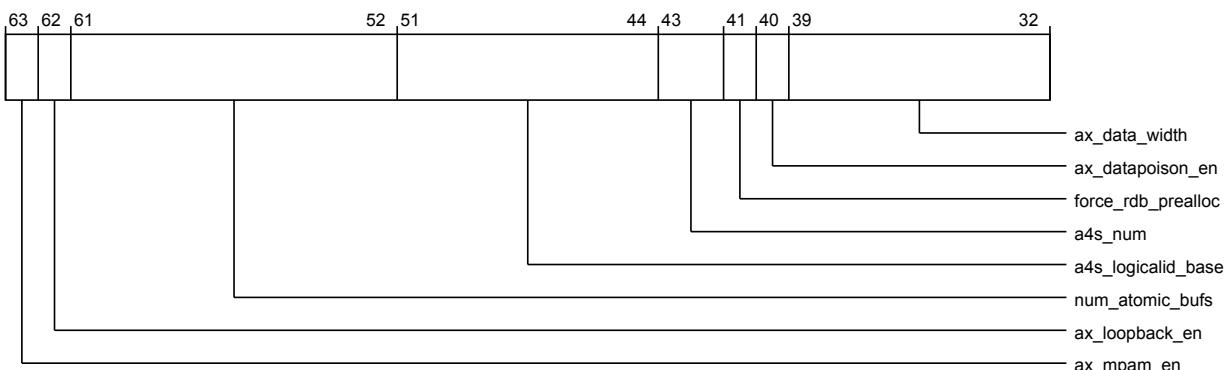


Figure 3-1029 por_rni_por_rni_unit_info (high)

The following table shows the por_rni_unit_info higher register bit assignments.

Table 3-1049 por_rni_por_rni_unit_info (high)

Bits	Field name	Description	Type	Reset
63	ax_mpam_en	MPAM enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
62	ax_loopback_en	LoopBack enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
61:52	num_atomic_bufs	Number of atomic data buffers	RO	Configuration dependent
51:44	a4s_logicalid_base	AXI4Stream interfaces logical ID base	RO	Configuration dependent
43:42	a4s_num	Number of AXI4Stream interfaces present	RO	Configuration dependent
41	force_rdb_prealloc	Force read data buffer preallocation 1'b1: yes 1'b0: no	RO	Configuration dependent
40	ax_datapoison_en	Data Poison enable on ACE-Lite/AXI4 interface 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent
39:32	ax_data_width	AXI interface data width in bits	RO	Configuration dependent

The following image shows the lower register bit assignments.

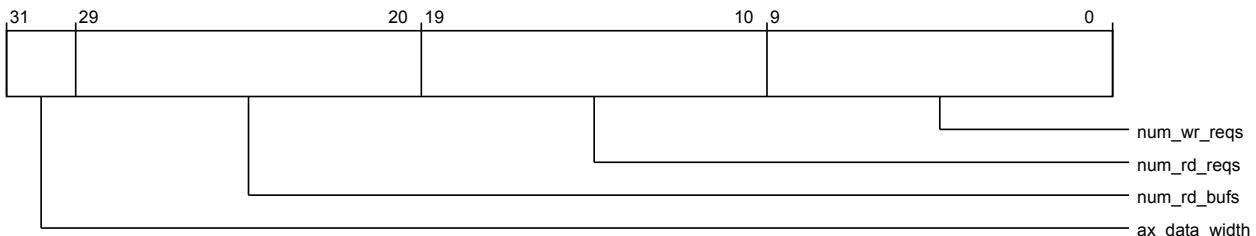


Figure 3-1030 por_rni_por_rni_unit_info (low)

The following table shows the por_rni_unit_info lower register bit assignments.

Table 3-1050 por_rni_por_rni_unit_info (low)

Bits	Field name	Description	Type	Reset
31:30	ax_data_width	AXI interface data width in bits	RO	Configuration dependent
29:20	num_rd_bufs	Number of read data buffers	RO	Configuration dependent
19:10	num_rd_reqs	Number of outstanding read requests	RO	Configuration dependent
9:0	num_wr_reqs	Number of outstanding write requests	RO	Configuration dependent

por_rni_unit_info2

Provides additional component identification information for RN-I.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h908
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1031 por_rni_por_rni_unit_info2 (high)

The following table shows the por_rni_unit_info2 higher register bit assignments.

Table 3-1051 por_rni_por_rni_unit_info2 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

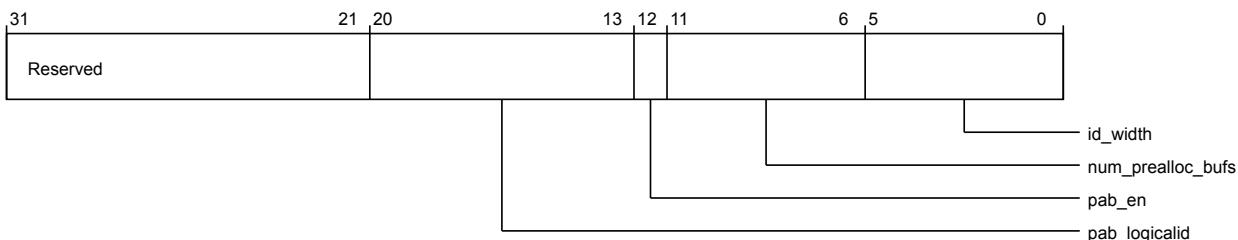


Figure 3-1032 por_rni_por_rni_unit_info2 (low)

The following table shows the por_rni_unit_info2 lower register bit assignments.

Table 3-1052 por_rni_por_rni_unit_info2 (low)

Bits	Field name	Description	Type	Reset
31:21	Reserved	Reserved	RO	-
20:13	pab_logicalid	PUB AUB bridge Logical ID	RO	Configuration dependent
12	pab_en	PUB AUB bridge enable 1'b1: Enabled 1'b0: Not enabled	RO	Configuration dependent

Table 3-1052 por_rni_por_rni_unit_info2 (low) (continued)

Bits	Field name	Description	Type	Reset
11:6	num_prealloc_bufs	Number of Pre-allocated Read Data Buffers	RO	Configuration dependent
5:0	id_width	AXI ID width for ACE-Lite slave ports	RO	Configuration dependent

por_rni_cfg_ctl

Functions as the configuration control register. Specifies the current mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rni_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

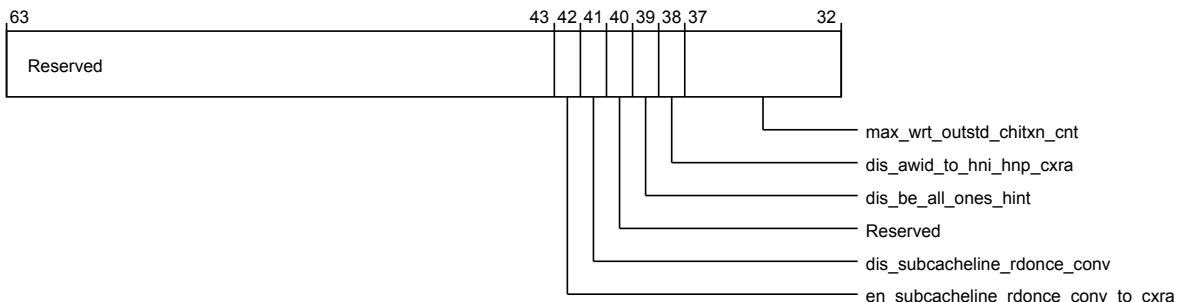


Figure 3-1033 por_rni_por_rni_cfg_ctl (high)

The following table shows the por_rni_cfg_ctl higher register bit assignments.

Table 3-1053 por_rni_por_rni_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:43	Reserved	Reserved	RO	-
42	en_subcacheline_rdonce_conv_to_cxra	If set, enables the conversion of sub-cacheline RdOnce to RdNoSnp for CXRA targets	RW	1'b0
41	dis_subcacheline_rdonce_conv	If set, disables the conversion of sub-cacheline RdOnce to RdNoSnp across all targets	RW	1'b0
40	Reserved	Reserved	RO	-
39	dis_be_all_ones_hint	If set, disables hint to HNF which signals all BE=1's on writes	RW	1'b0

Table 3-1053 por_rni_por_rni_cfg_ctl (high) (continued)

Bits	Field name	Description	Type	Reset
38	dis_awid_to_hni_hnp_cxra	If set, disables compressed AWID to HNI, HNP and CXRA, also disables compressed AWID based ordering. Set this bit if uniq-ID write performance is needed.	RW	1'b0
37:32	max_wrt_outstd_chitxn_cnt	Maximum number of outstanding writes allowed on CHI-side	RW	Configuration dependent

The following image shows the lower register bit assignments.

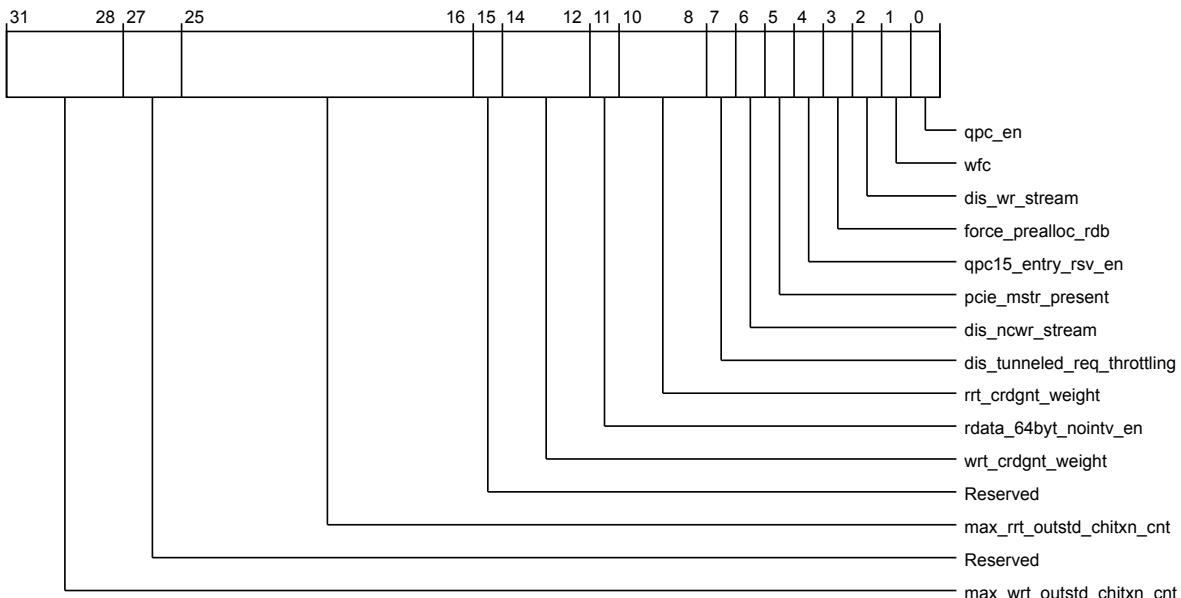


Figure 3-1034 por_rni_por_rni_cfg_ctl (low)

The following table shows the por_rni_cfg_ctl lower register bit assignments.

Table 3-1054 por_rni_por_rni_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	max_wrt_outstd_chitxn_cnt	Maximum number of outstanding writes allowed on CHI-side	RW	Configuration dependent
27:26	Reserved	Reserved	RO	-
25:16	max_rrt_outstd_chitxn_cnt	Maximum number of outstanding reads allowed on CHI-side	RW	Configuration dependent
15	Reserved	Reserved	RO	-
14:12	wrt_crdgnt_weight	Determines weight of credit grant allocated to retried writes in presence of pending retried reads	RW	3'b001
11	rdata_64byt_nointv_en	Enables no interleaving property on normal memory read data within 64B granule when set	RW	1'b1

Table 3-1054 por_rni_por_rni_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
10:8	rrt_crdgnt_weight	Determines weight of credit grant allocated to retried reads in presence of pending retried writes	RW	3'b100
7	dis_tunneled_req_throttling	Disables retry based throttling of tunneled write requests	RW	1'b0
6	dis_ncwr_stream	Disables streaming of ordered non-cacheable writes when set	RW	1'b0
5	pcie_mstr_present	Indicates PCIe master is present; must be set if PCIe master is present upstream of RN-I or RN-D	RW	1'b0
4	qpc15_entry_rsv_en	Enables QPC15 entry reservation 1'b1: Reserves tracker entry for QoS15 requests 1'b0: Does not reserve tracker entry for QoS15 requests NOTE: Only valid and applicable when por_rnd_qpc_en is set	RW	1'b0
3	force_prealloc_rdb	When set, all reads from the RN-I are sent with a preallocated read data buffer	RW	Configuration dependent
2	dis_wr_stream	Disables streaming of ordered writes when set	RW	1'b0
1	wfc	When set, enables waiting for completion (COMP) before dispatching dependent transaction (TXN)	RW	1'b0
0	qpc_en	When set, enables QPC-based scheduling using two QoS priority classes (QoS15 and non-QoS15)	RW	1'b1

por_rni_aux_ctl

Functions as the auxiliary control register for RN-I.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

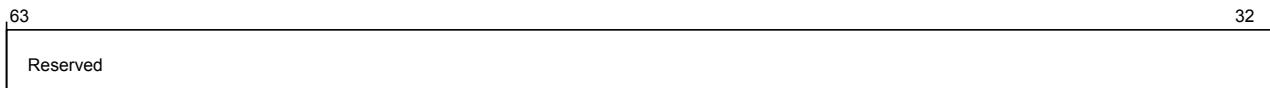


Figure 3-1035 por_rni_por_rni_aux_ctl (high)

The following table shows the por_rni_aux_ctl higher register bit assignments.

Table 3-1055 por_rni_por_rni_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

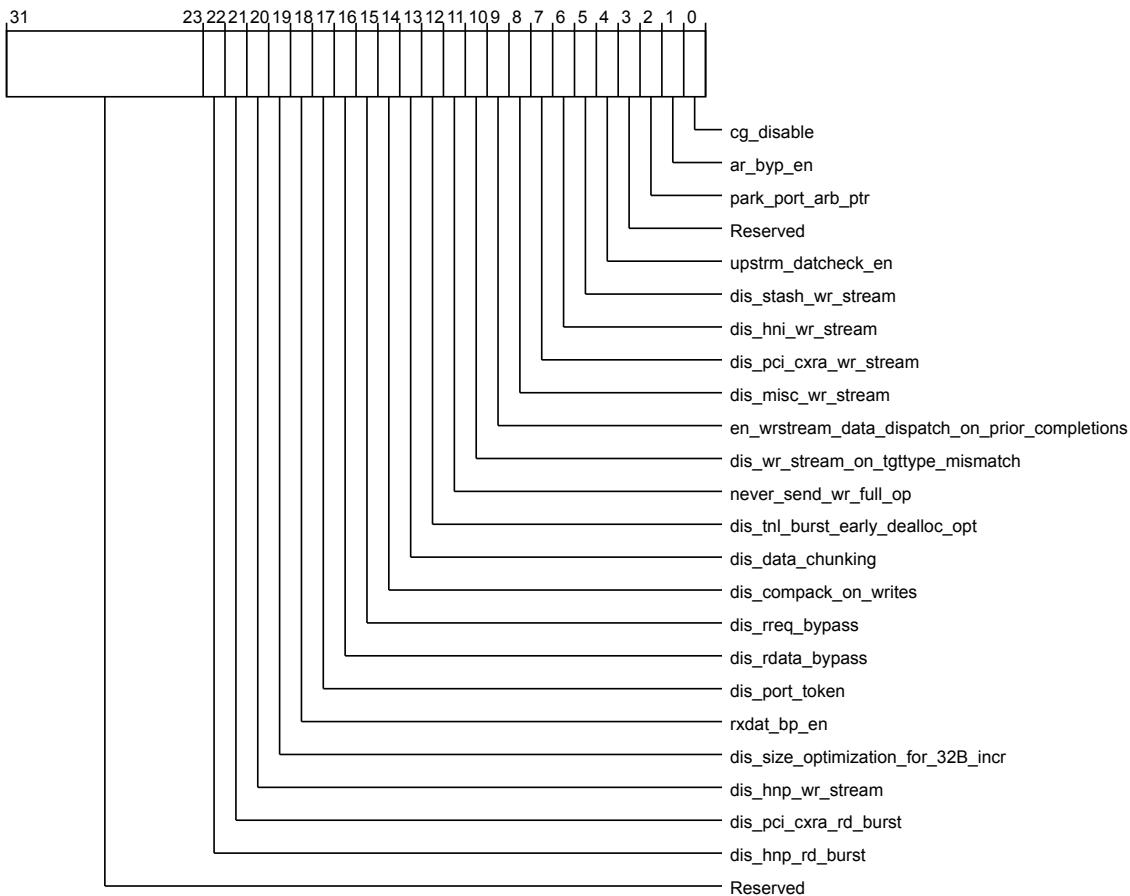


Figure 3-1036 por_rni_por_rni_aux_ctl (low)

The following table shows the por_rni_aux_ctl lower register bit assignments.

Table 3-1056 por_rni_por_rni_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:23	Reserved	Reserved	RO	-
22	dis_hnp_rd_burst	If set, disables read burst to HNP on CHI request flits. Read burst on CHI is supported only in non-decoupled RDB configuration.	RW	1'b0
21	dis_pci_cxra_rd_burst	If set, disables read burst to PCI-CXRA on CHI request flits. Read burst on CHI is supported only in non-decoupled RDB configuration.	RW	1'b0

Table 3-1056 por_rni_por_rni_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
20	dis_hnp_wr_stream	Disables streaming of ordered writes to HNP when set	RW	1'b0
19	dis_size_optimization_for_32B_incr	If set, disables the size related optimization for a 32B INCR burst (rh-2512)	RW	1'b0
18	rxdat_bp_en	If set, back pressures the rxdat interface when RDB's are not available	RW	1'b0
17	dis_port_token	If set, disables per port reservation in the tracker(rd and wr)	RW	1'b1
16	dis_rdata_bypass	If set, disables read data bypass path; if rxdat_bp_en = 1'b0	RW	1'b0
15	dis_rreq_bypass	If set, disables read request bypass path	RW	1'b0
14	dis_compack_on_writes	If set, disables comp_ack on streaming writes. WrData is used for ordering writes	RW	1'b1
13	dis_data_chunking	If set, disables the data chunking feature	RW	1'b0
12	dis_tnl_burst_early_dealloc_opt	If set, disables the optimization related to early deallocation of tunneled writes for intermediate txns of burst	RW	1'b0
11	never_send_wr_full_op	If set, RNI will never send WR FULL op. All write ops will be of PTL type	RW	1'b0
10	dis_wr_stream_on_tgtype_mismatch	If set, serializes first write when moving from one tgtype to another	RW	1'b0
9	en_wrstream_data_dispatch_on_prior_completions	If set, data dispatch for streaming writes waits for completion of all older writes	RW	1'b0
8	dis_misc_wr_stream	Disables streaming of ordered writes with following attributes when set : Device memory or EWA=0 or Excl=1	RW	1'b0
7	dis_pci_cxra_wr_stream	Disables streaming of ordered writes to PCI-CXRA when set	RW	1'b0
6	dis_hni_wr_stream	Disables streaming of ordered writes to HNI when set	RW	1'b0
5	dis_stash_wr_stream	Disables streaming of ordered WrUniqStash when set	RW	1'b0
4	upstrm_datcheck_en	Upstream supports Datacheck	RW	Configuration dependent
3	Reserved	Reserved	RO	-

Table 3-1056 por_rni_por_rni_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
2	park_port_arb_ptr	Parks the AXI port arbitration pointer for Burst	RW	1'b0
1	ar_byp_en	AR bypass enable; enables bypass path in the AR pipeline	RW	1'b1
0	cg_disable	Disables clock gating when set	RW	1'b0

por_rni_s_0-2_port_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls port S#{index} AXI/ACE slave interface settings.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA10 + # {[0, 1, 2]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rni_secure_register_groups_override.port_ctrl

The following image shows the higher register bit assignments.



Figure 3-1037 por_rni_por_rni_s_0-2_port_control (high)

The following table shows the por_rni_s_0-2_port_control higher register bit assignments.

Table 3-1057 por_rni_por_rni_s_0-2_port_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

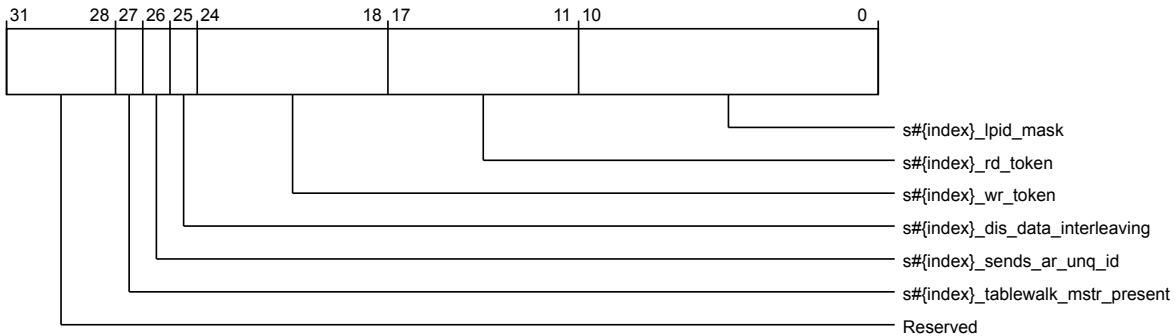


Figure 3-1038 por_rni_por_rni_s_0-2_port_control (low)

The following table shows the por_rni_s_0-2_port_control lower register bit assignments.

Table 3-1058 por_rni_por_rni_s_0-2_port_control (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27	s#{index}_tablewalk_mstr_present	If set, Indicates translation table walk master present such as TCU or GIC	RW	1'b0
26	s#{index}_sends_ar_unq_id	If set, indicates AR transactions on Port#{index} are always Unique ID. This bit for a port must be set to 1 to enable Read Burst on the CHI side of RND. This bit works in conjunction with dis_rd_burst being 0.	RW	1'b0
25	s#{index}_dis_data_interleaving	If set, disables read DATA interleaving on RDATAS#{index} channel. This applies only to RDATA generated as a response to requests on AR channel . This does not apply to RDATA generated as a response to Atomic request on AW channel. I.e. RDATA of an Atomic op, on AW channel, may interleave with RDATA of an AR channel request	RW	1'b0
24:18	s#{index}_wr_token	Port S#{index} reserved token count for AW channel This must be less than the number of Wr requests(RNID_NUM_XRT_REQ) on AW achnnel	RW	6'b00_0000

Table 3-1058 por_rni_por_rni_s_0-2_port_control (low) (continued)

Bits	Field name	Description	Type	Reset
17:11	s#{index}_rd_token	Port S#{index} reserved token count for AR channel per slice This should be less than the number of Rd requests(RNID_NUM_XRT_SLICE_REQ) per slice on AR achnnel	RW	6'b00_0000
10:0	s#{index}_lpid_mask	Port S#{index} LPID mask LPID[0]: Equal to the result of UnaryOR of BitwiseAND of LPID mask and AXID (LPID[0] = (AXID and mask)); specifies which AXID bit is reflected in the LSB of LPID LPID[2:1]: Equal to port ID[1:0]; the MSB of LPID contains port ID	RW	11'b000_0000_0000

por_rni_s_0-2_mpam_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls port S#{index} AXI/ACE slave interface MPAM override values

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA28 + # {[0, 1, 2]*8}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rni_secure_register_groups_override.mpam_ctrl

The following image shows the higher register bit assignments.



Figure 3-1039 por_rni_por_rni_s_0-2_mpam_control (high)

The following table shows the por_rni_s_0-2_mpam_control higher register bit assignments.

Table 3-1059 por_rni_por_rni_s_0-2_mpam_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

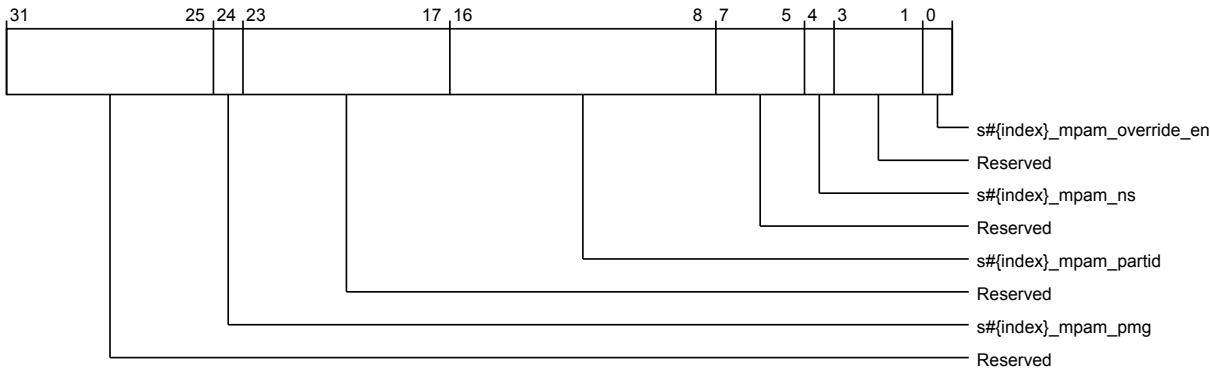


Figure 3-1040 por_rni_por_rni_s_0-2_mpam_control (low)

The following table shows the por_rni_s_0-2_mpam_control lower register bit assignments.

Table 3-1060 por_rni_por_rni_s_0-2_mpam_control (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	s#{index}_mpam_pmg	Port S#{index} MPAM_PMG value	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	s#{index}_mpam_partid	Port S#{index} MPAM_PARTID value	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	s#{index}_mpam_ns	Port S#{index} MPAM_NS value	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	s#{index}_mpam_override_en	Port S#{index} MPAM override en When set, MPAM value on CHI side is driven from MPAM override value in this register. Note that when RNID_AXMPAM_EN_PARAM is set to 0, MPAM override value is always used irrespective of this bit value	RW	1'b0

por_rni_s_0-2_qos_control

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls QoS settings for port S#{index} AXI/ACE slave interface.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA80 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

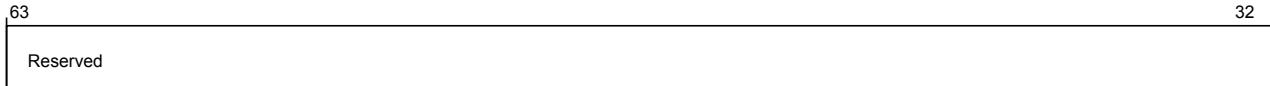


Figure 3-1041 por_rni_por_rni_s_0-2_qos_control (high)

The following table shows the por_rni_s_0-2_qos_control higher register bit assignments.

Table 3-1061 por_rni_por_rni_s_0-2_qos_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

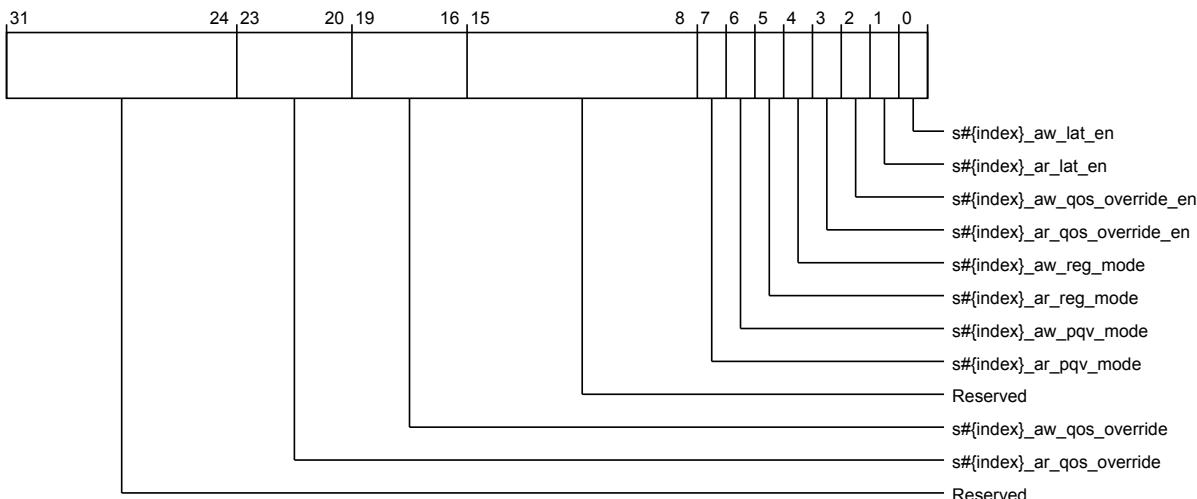


Figure 3-1042 por_rni_por_rni_s_0-2_qos_control (low)

The following table shows the por_rni_s_0-2_qos_control lower register bit assignments.

Table 3-1062 por_rni_por_rni_s_0-2_qos_control (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:20	s#{index}_ar_qos_override	AR QoS override value for port S#{index}	RW	4'b0000
19:16	s#{index}_aw_qos_override	AW QoS override value for port S#{index}	RW	4'b0000
15:8	Reserved	Reserved	RO	-

Table 3-1062 por_rni_por_rni_s_0-2_qos_control (low) (continued)

Bits	Field name	Description	Type	Reset
7	s#{index}_ar_pqv_mode	Configures the QoS regulator mode for read transactions during period mode 1'b0: Normal mode; QoS value is stable when the master is idle 1'b1: Quiesce high mode; QoS value tends to the maximum value when the master is idle	RW	1'b0
6	s#{index}_aw_pqv_mode	Configures the QoS regulator mode for write transactions during period mode 1'b0: Normal mode; QoS value is stable when the master is idle 1'b1: Quiesce high mode; QoS value tends to the maximum value when the master is idle	RW	1'b0
5	s#{index}_ar_reg_mode	Configures the QoS regulator mode for read transactions 1'b0: Latency mode 1'b1: Period mode; used for bandwidth regulation	RW	1'b0
4	s#{index}_aw_reg_mode	Configures the QoS regulator mode for write transactions 1'b0: Latency mode 1'b1: Period mode; used for bandwidth regulation	RW	1'b0
3	s#{index}_ar_qos_override_en	Enables port S#{index} AR QoS override; when set, allows QoS value on inbound AR transactions to be overridden	RW	1'b0
2	s#{index}_aw_qos_override_en	Enables port S#{index} AW QoS override; when set, allows QoS value on inbound AW transactions to be overridden	RW	1'b0
1	s#{index}_ar_lat_en	Enables port S#{index} AR QoS regulation when set	RW	1'b0
0	s#{index}_aw_lat_en	Enables port S#{index} AW QoS regulation when set	RW	1'b0

por_rni_s_0-2_qos_lat_tgt

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls QoS target latency (in cycles) for regulations of port S#{index} read and write transactions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA88 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

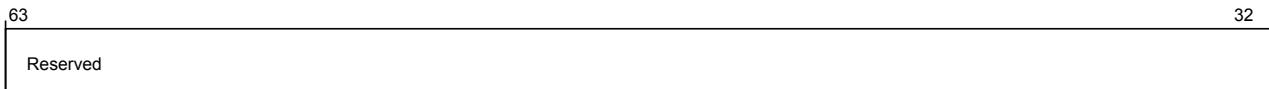


Figure 3-1043 por_rni_por_rni_s_0-2_qos_lat_tgt (high)

The following table shows the por_rni_s_0-2_qos_lat_tgt higher register bit assignments.

Table 3-1063 por_rni_por_rni_s_0-2_qos_lat_tgt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

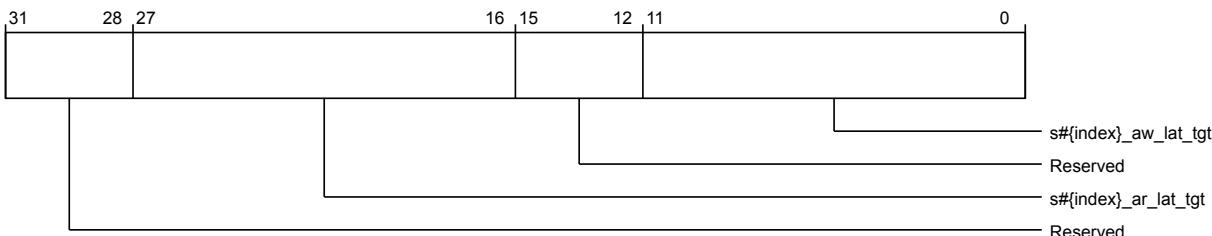


Figure 3-1044 por_rni_por_rni_s_0-2_qos_lat_tgt (low)

The following table shows the por_rni_s_0-2_qos_lat_tgt lower register bit assignments.

Table 3-1064 por_rni_por_rni_s_0-2_qos_lat_tgt (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:16	s#{index}_ar_lat_tgt	Port S#{index} AR channel target latency; a value of 0 corresponds to no regulation	RW	12'h000
15:12	Reserved	Reserved	RO	-
11:0	s#{index}_aw_lat_tgt	Port S#{index} AW channel target latency; a value of 0 corresponds to no regulation	RW	12'h000

por_rni_s_0-2_qos_lat_scale

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls the QoS target latency scale factor for port S#{index} read and write transactions. This register represents powers of two from the range 2^{-5} to 2^{-12} ; it is used to match a 16-bit integrator.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA90 + # {[0, 1, 2]*32}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.

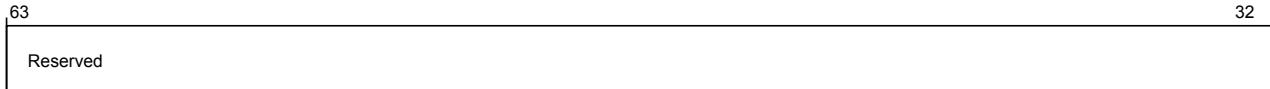


Figure 3-1045 por_rni_por_rni_s_0-2_qos_lat_scale (high)

The following table shows the por_rni_s_0-2_qos_lat_scale higher register bit assignments.

Table 3-1065 por_rni_por_rni_s_0-2_qos_lat_scale (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

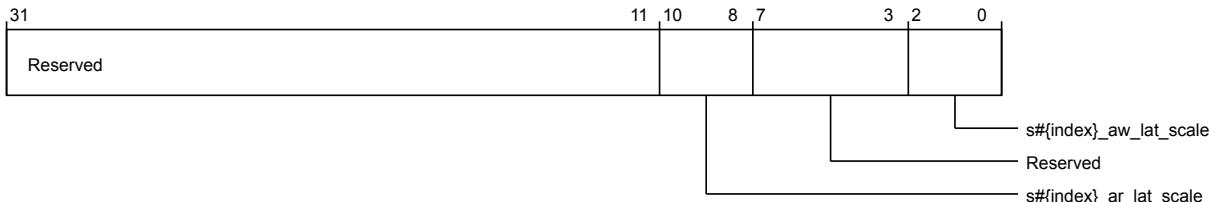


Figure 3-1046 por_rni_por_rni_s_0-2_qos_lat_scale (low)

The following table shows the por_rni_s_0-2_qos_lat_scale lower register bit assignments.

Table 3-1066 por_rni_por_rni_s_0-2_qos_lat_scale (low)

Bits	Field name	Description	Type	Reset
31:11	Reserved	Reserved	RO	-
10:8	s#{index}_ar_lat_scale	Port S#{index} AR QoS scale factor 3'b000: 2 ⁽⁻⁵⁾ 3'b001: 2 ⁽⁻⁶⁾ 3'b010: 2 ⁽⁻⁷⁾ 3'b011: 2 ⁽⁻⁸⁾ 3'b100: 2 ⁽⁻⁹⁾ 3'b101: 2 ⁽⁻¹⁰⁾ 3'b110: 2 ⁽⁻¹¹⁾ 3'b111: 2 ⁽⁻¹²⁾	RW	3'h0

Table 3-1066 por_rni_por_rni_s_0-2_qos_lat_scale (low) (continued)

Bits	Field name	Description	Type	Reset
7:3	Reserved	Reserved	RO	-
2:0	s#{index}_aw_lat_scale	Port S#{index} AW QoS scale factor 3'b000: 2 ⁽⁻⁵⁾ 3'b001: 2 ⁽⁻⁶⁾ 3'b010: 2 ⁽⁻⁷⁾ 3'b011: 2 ⁽⁻⁸⁾ 3'b100: 2 ⁽⁻⁹⁾ 3'b101: 2 ⁽⁻¹⁰⁾ 3'b110: 2 ⁽⁻¹¹⁾ 3'b111: 2 ⁽⁻¹²⁾	RW	3'h0

por_rni_s_0-2_qos_lat_range

This register repeats 2 times. It parametrized by the index from 0 to 2. Controls the minimum and maximum QoS values generated by the QoS latency regulator for port S#{index} read and write transactions.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA98 + # {[0, 1, 2]*32}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_rni_secure_register_groups_override.qos_ctrl

The following image shows the higher register bit assignments.



Figure 3-1047 por_rni_por_rni_s_0-2_qos_lat_range (high)

The following table shows the por_rni_s_0-2_qos_lat_range higher register bit assignments.

Table 3-1067 por_rni_por_rni_s_0-2_qos_lat_range (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

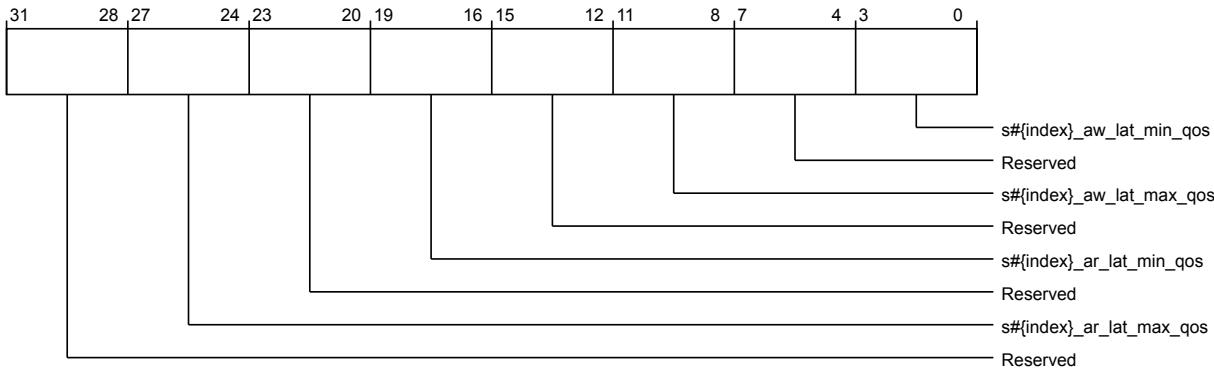


Figure 3-1048 por_rni_por_rni_s_0-2_qos_lat_range (low)

The following table shows the por_rni_s_0-2_qos_lat_range lower register bit assignments.

Table 3-1068 por_rni_por_rni_s_0-2_qos_lat_range (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:24	s#{index}_ar_lat_max_qos	Port S#{index} AR QoS maximum value	RW	4'h0
23:20	Reserved	Reserved	RO	-
19:16	s#{index}_ar_lat_min_qos	Port S#{index} AR QoS minimum value	RW	4'h0
15:12	Reserved	Reserved	RO	-
11:8	s#{index}_aw_lat_max_qos	Port S#{index} AW QoS maximum value	RW	4'h0
7:4	Reserved	Reserved	RO	-
3:0	s#{index}_aw_lat_min_qos	Port S#{index} AW QoS minimum value	RW	4'h0

por_rni_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

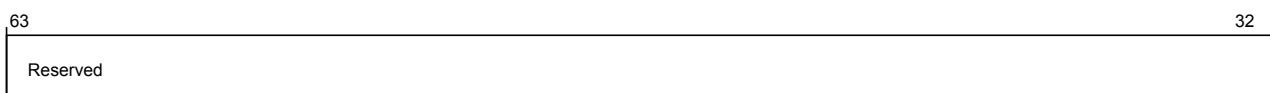


Figure 3-1049 por_rni_por_rni_pmu_event_sel (high)

The following table shows the por_rni_pmu_event_sel higher register bit assignments.

Table 3-1069 por_rni_por_rni_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

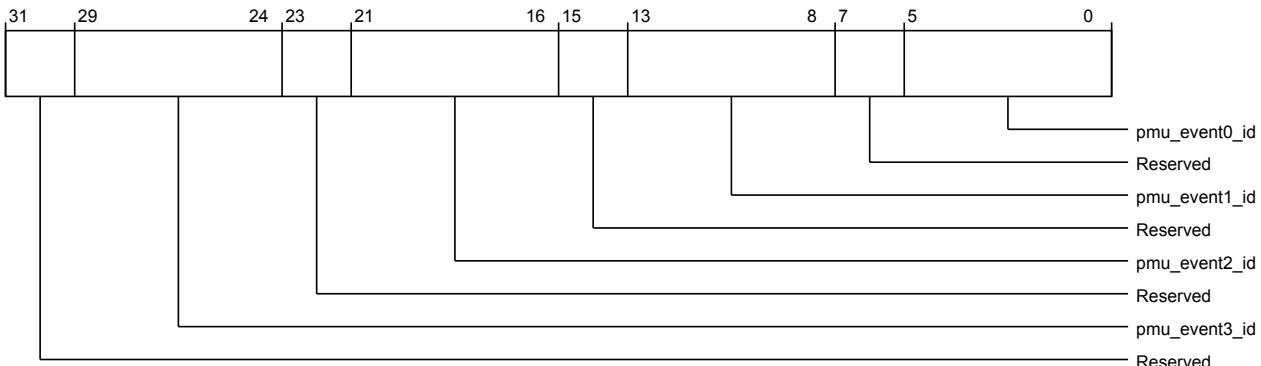


Figure 3-1050 por_rni_por_rni_pmu_event_sel (low)

The following table shows the por_rni_pmu_event_sel lower register bit assignments.

Table 3-1070 por_rni_por_rni_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	RN-I PMU Event 3 ID; see pmu_event0_id for encodings	RW	6'b0
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	RN-I PMU Event 2 ID; see pmu_event0_id for encodings	RW	6'b0
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	RN-I PMU Event 1 ID; see pmu_event0_id for encodings	RW	6'b0

Table 3-1070 por_rni_por_rni_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-

Table 3-1070 por_rni_por_rni_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
5:0	pmu_event0_id	RN-I PMU Event 0 ID 6'h00: No event 6'h01: Port S0 RDataBeats 6'h02: Port S1 RDataBeats 6'h03: Port S2 RDataBeats 6'h04: RXDAT flits received 6'h05: TXDAT flits sent 6'h06: Total TXREQ flits sent 6'h07: Retried TXREQ flits sent 6'h08: RRT occupancy count overflow 6'h09: WRT occupancy count overflow 6'h0A: Replayed TXREQ flits 6'h0B: WriteCancel sent 6'h0C: Port S0 WDataBeats 6'h0D: Port S1 WDataBeats 6'h0E: Port S2 WDataBeats 6'h0F: RRT allocation 6'h10: WRT allocation 6'h11: PADB occupancy count overflow 6'h12: RPDB occupancy count overflow 6'h13: RRT occupancy count overflow_slice1 6'h14: RRT occupancy count overflow_slice2 6'h15: RRT occupancy count overflow_slice3 6'h16: WRT request throttled 6'h17: RNI backpressure CHI LDB full 6'h18: RRT normal rd req occupancy count overflow_slice0 6'h19: RRT normal rd req occupancy count overflow_slice1 6'h1A: RRT normal rd req occupancy count overflow_slice2 6'h1B: RRT normal rd req occupancy count overflow_slice3 6'h1C: RRT PCIe RD burst req occupancy count overflow_slice0 6'h1D: RRT PCIe RD burst req occupancy count overflow_slice1 6'h1E: RRT PCIe RD burst req occupancy count overflow_slice2 6'h1F: RRT PCIe RD burst req occupancy count overflow_slice3 6'h20: RRT PCIe RD burst allocation 6'h21: Compressed AWID ordering 6'h22: Atomic data buffer allocation 6'h23: Atomic data buffer occupancy	RW	6'b0

3.3.18 XP register descriptions

This section lists the XP registers.

por_mxp_node_info

Provides component identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h0
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

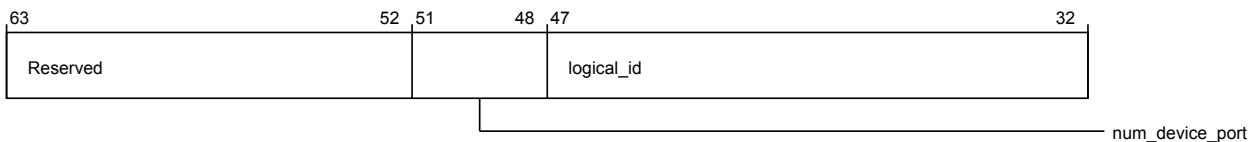


Figure 3-1051 por_mxp_por_mxp_node_info (high)

The following table shows the por_mxp_node_info higher register bit assignments.

Table 3-1071 por_mxp_por_mxp_node_info (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:48	num_device_port	Number of device ports attached to the MXP. Mesh config = (1x1)? Max. of 6 Device Ports are supported : Max. of 4 Device Ports are supported	RO	Configuration dependent
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

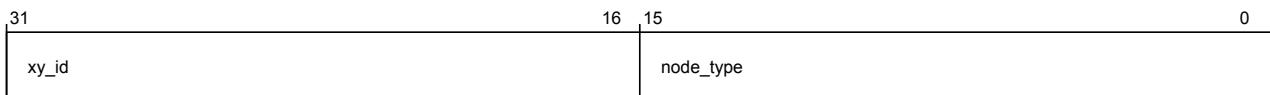


Figure 3-1052 por_mxp_por_mxp_node_info (low)

The following table shows the por_mxp_node_info lower register bit assignments.

Table 3-1072 por_mxp_por_mxp_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	xy_id	Identifies (X,Y) location of XP within the mesh NOTE: The (X,Y) location is specified following the node ID format as defined in Node ID mapping section, with the bottom 3 bits, corresponding to port ID and device ID, set to 0. Bits 31:11 must always be set to 0. The range of bits representing the (X,Y) location varies for different node ID formats.	RO	16'h0000
15:0	node_type	CMN-600 node type identifier	RO	16'h0006

por_mxp_device_port_connect_info_p_0-5

This register repeats 5 times. It parametrized by the index from 0 to 5. Contains device port connection information for port # {index}.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h8 + # {8*[0, 1, 2, 3, 4, 5]}
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

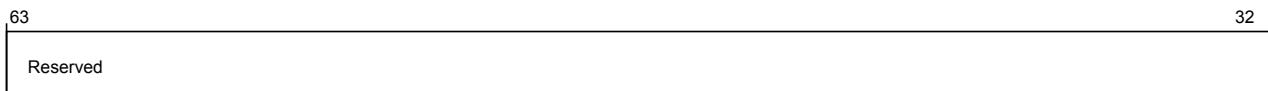


Figure 3-1053 por_mxp_por_mxp_device_port_connect_info_p_0-5 (high)

The following table shows the por_mxp_device_port_connect_info_p_0-5 higher register bit assignments.

Table 3-1073 por_mxp_por_mxp_device_port_connect_info_p_0-5 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

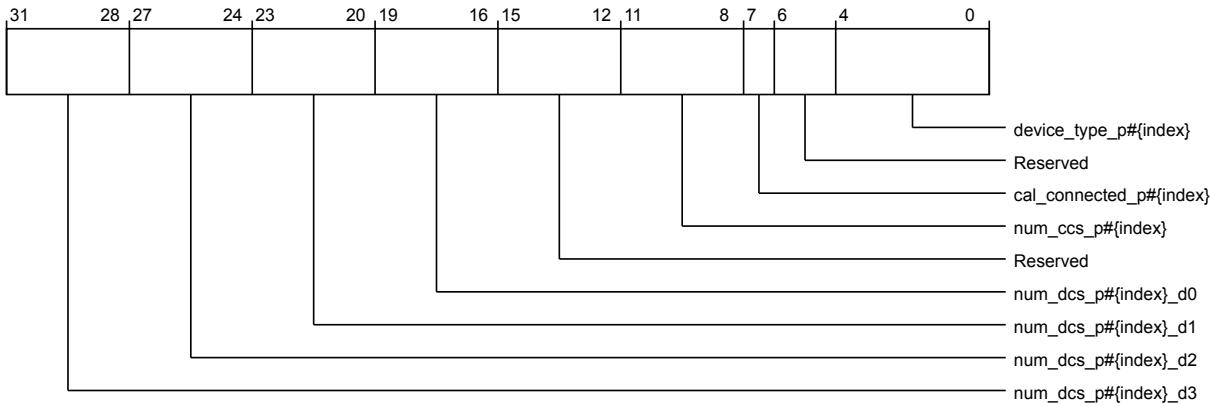


Figure 3-1054 por_mxp_por_mxp_device_port_connect_info_p_0-5 (low)

The following table shows the por_mxp_device_port_connect_info_p_0-5 lower register bit assignments.

Table 3-1074 por_mxp_por_mxp_device_port_connect_info_p_0-5 (low)

Bits	Field name	Description	Type	Reset
31:28	num_dcs_p#{index}_d3	Number of device credited slices connected to port #{index} device 3 (Allowed values: 0-4)	RO	Configuration dependent
27:24	num_dcs_p#{index}_d2	Number of device credited slices connected to port #{index} device 2 (Allowed values: 0-4)	RO	Configuration dependent
23:20	num_dcs_p#{index}_d1	Number of device credited slices connected to port #{index} device 1 (Allowed values: 0-4)	RO	Configuration dependent
19:16	num_dcs_p#{index}_d0	Number of device credited slices connected to port #{index} device 0 (Allowed values: 0-4)	RO	Configuration dependent
15:12	Reserved	Reserved	RO	-
11:8	num_ccs_p#{index}	Number of CAL credited slices connected to port #{index} (Allowed values: 0-2)	RO	Configuration dependent
7	cal_connected_p#{index}	When set, CAL is connected on port #{index} (Allowed values: 0-1)	RO	(MXP_NUM_DEV_P#{index}_PARAM > 1) ? 1'b1 : 1'b0

Table 3-1074 por_mxp_por_mxp_device_port_connect_info_p_0-5 (low) (continued)

Bits	Field name	Description	Type	Reset
6:5	Reserved	Reserved	RO	-
4:0	device_type_p#{index}	Connected device type 5'b00000: Reserved 5'b00001: RN-I 5'b00010: RN-D 5'b00011: Reserved 5'b00100: RN-F_CHIB 5'b00101: RN-F_CHIB_ESAM 5'b00110: RN-F_CHIA 5'b00111: RN-F_CHIA_ESAM 5'b01000: HN-T 5'b01001: HN-I 5'b01010: HN-D 5'b01011: HN-P 5'b01100: SN-F_CHIC 5'b01101: SBSX 5'b01110: HN-F 5'b01111: SN-F_CHIE 5'b10000: SN-F_CHID 5'b10001: CXHA 5'b10010: CXRA 5'b10011: CXRH 5'b10100: RN-F_CHID 5'b10101: RN-F_CHID_ESAM 5'b10110: RN-F_CHIC 5'b10111: RN-F_CHIC_ESAM 5'b11000: RN-F_CHIE 5'b11001: RN-F_CHIE_ESAM 5'b11010: Reserved 5'b11011: Reserved 5'b11100: MTSX 5'b11101: HN-V 5'b11110: CCG 5'b11111: Reserved	RO	Configuration dependent

por_mxp_mesh_port_connect_info_east

Contains port connection information for East port.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h38
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

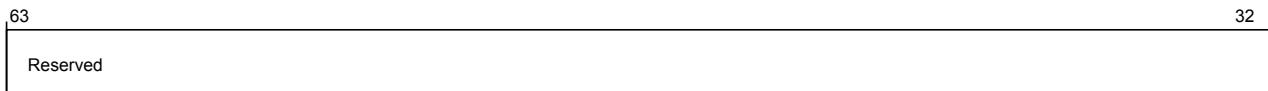


Figure 3-1055 por_mxp_por_mxp_mesh_port_connect_info_east (high)

The following table shows the por_mxp_mesh_port_connect_info_east higher register bit assignments.

Table 3-1075 por_mxp_por_mxp_mesh_port_connect_info_east (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

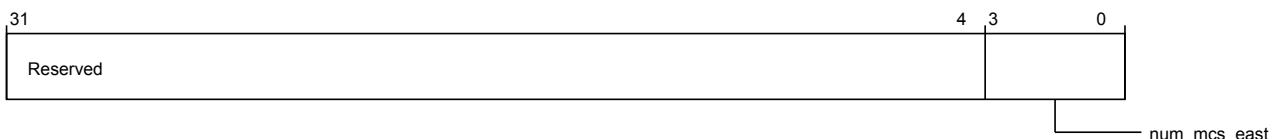


Figure 3-1056 por_mxp_por_mxp_mesh_port_connect_info_east (low)

The following table shows the por_mxp_mesh_port_connect_info_east lower register bit assignments.

Table 3-1076 por_mxp_por_mxp_mesh_port_connect_info_east (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	num_mcs_east	Number of mesh credited slices connected to East port (Allowed values: 0-4)	RO	Configuration dependent

por_mxp_mesh_port_connect_info_north

Contains port connection information for North port.

Its characteristics are:

Type	RO
Register width (Bits)	64

Address offset	16'h40
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

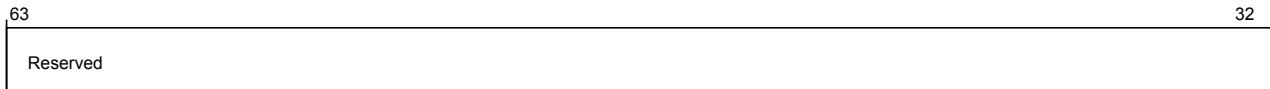


Figure 3-1057 por_mxp_por_mxp_mesh_port_connect_info_north (high)

The following table shows the por_mxp_mesh_port_connect_info_north higher register bit assignments.

Table 3-1077 por_mxp_por_mxp_mesh_port_connect_info_north (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

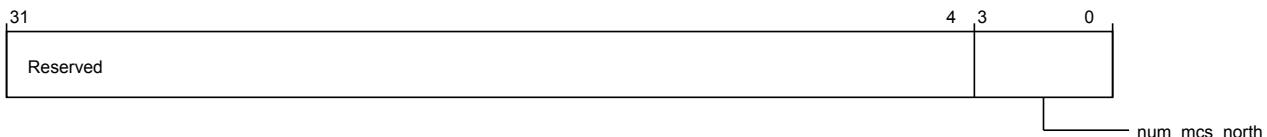


Figure 3-1058 por_mxp_por_mxp_mesh_port_connect_info_north (low)

The following table shows the por_mxp_mesh_port_connect_info_north lower register bit assignments.

Table 3-1078 por_mxp_por_mxp_mesh_port_connect_info_north (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	num_mcs_north	Number of mesh credited slices connected to North port (Allowed values: 0-4)	RO	Configuration dependent

por_mxp_device_port_connect_Idid_info_p_0-5

This register repeats 5 times. It parametrized by the index from 0 to 5. Contains LDID information for devices connected to port #{{index}}. Valid only for RNFs

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h48 + #{{0, 1, 2, 3, 4, 5}}
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

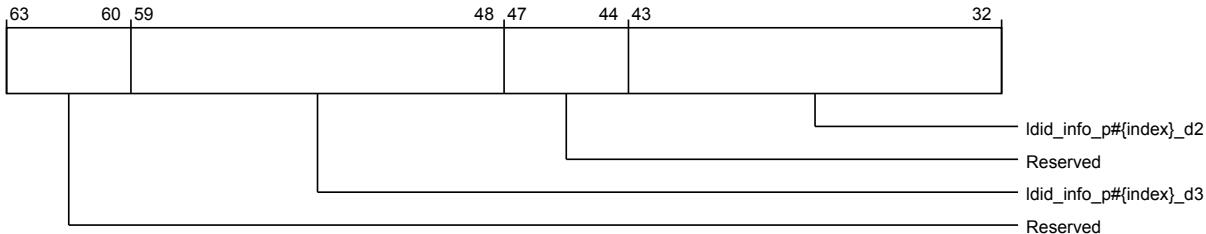


Figure 3-1059 por_mxp_port_connect_ldid_info_p_0-5 (high)

The following table shows the por_mxp_port_connect_ldid_info_p_0-5 higher register bit assignments.

Table 3-1079 por_mxp_port_connect_ldid_info_p_0-5 (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:48	ldid_info_p#{index}_d3	LDID value of the device connected to port P#{index}_D3	RO	Configuration dependent
47:44	Reserved	Reserved	RO	-
43:32	ldid_info_p#{index}_d2	LDID value of the device connected to port P#{index}_D2	RO	Configuration dependent

The following image shows the lower register bit assignments.

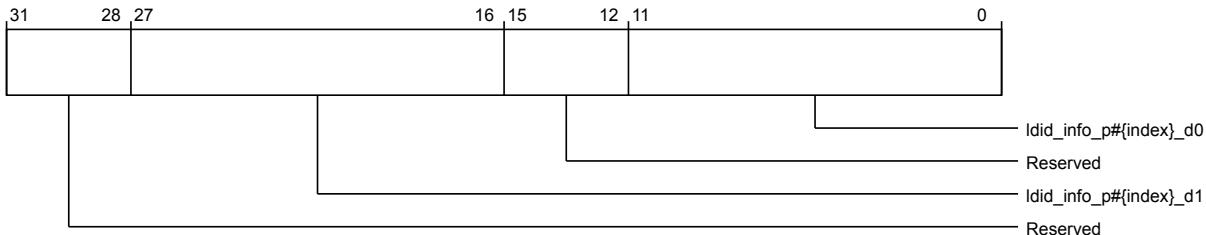


Figure 3-1060 por_mxp_port_connect_ldid_info_p_0-5 (low)

The following table shows the por_mxp_port_connect_ldid_info_p_0-5 lower register bit assignments.

Table 3-1080 por_mxp_port_connect_ldid_info_p_0-5 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:16	ldid_info_p#{index}_d1	LDID value of the device connected to port P#{index}_D1	RO	Configuration dependent
15:12	Reserved	Reserved	RO	-
11:0	ldid_info_p#{index}_d0	LDID value of the device connected to port P#{index}_D0	RO	Configuration dependent

por_mxp_child_info

Provides component child identification information.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h80
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1061 por_mxp_por_mxp_child_info (high)

The following table shows the por_mxp_child_info higher register bit assignments.

Table 3-1081 por_mxp_por_mxp_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

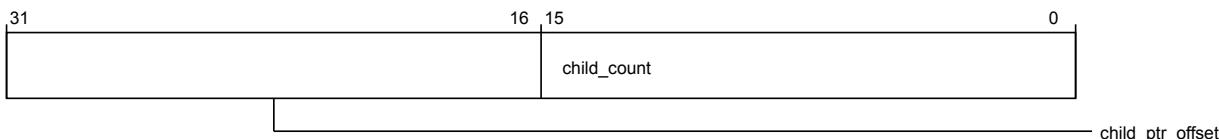


Figure 3-1062 por_mxp_por_mxp_child_info (low)

The following table shows the por_mxp_child_info lower register bit assignments.

Table 3-1082 por_mxp_por_mxp_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h100
15:0	child_count	Number of child nodes; used in discovery process	RO	Configuration dependent

por_mxp_child_pointer_0-31

This register repeats 31 times. It parametrized by the index from 0 to 31. Contains base address of the configuration slave for child # {index}.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h100 + # {8*[0, 1, 2, .., 30, 31]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

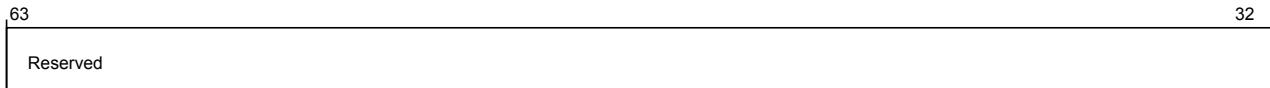


Figure 3-1063 por_mxp_por_mxp_child_pointer_0-31 (high)

The following table shows the por_mxp_child_pointer_0-31 higher register bit assignments.

Table 3-1083 por_mxp_por_mxp_child_pointer_0-31 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

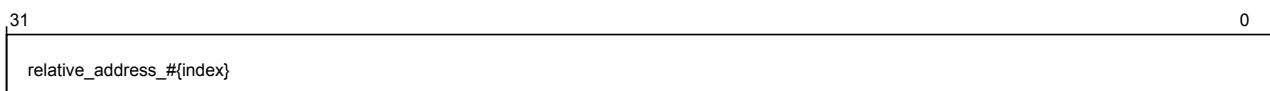


Figure 3-1064 por_mxp_por_mxp_child_pointer_0-31 (low)

The following table shows the por_mxp_child_pointer_0-31 lower register bit assignments.

Table 3-1084 por_mxp_por_mxp_child_pointer_0-31 (low)

Bits	Field name	Description	Type	Reset
31:0	relative_address_{index}	Bit [31]: External or internal child node 1'b1: Indicates this child pointer points to a configuration node that is external to CMN-600 1'b0: Indicates this child pointer points to a configuration node that is internal to CMN-600 Bits [30]: Set to 1'b0 Bits [29:0]: Child node address offset relative to PERIPHBASE	RO	32'b0

por_mxp_p_0-5_info

This register repeats 5 times. It parametrized by the index from 0 to 5. Provides component identification information for XP port # {index}. NOTE: There will be max. of 6 MXP Port Info registers based on MXP_NUM_DEV_PORT_PARAM value. Each successive MXP Port Info register will be at the next 8 byte address boundary. Each successive MXP Port Info register will be named with the suffix. For example por_mxp_p<0:5>_info

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900 + # {16*[0, 1, 2, 3, 4, 5]}

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

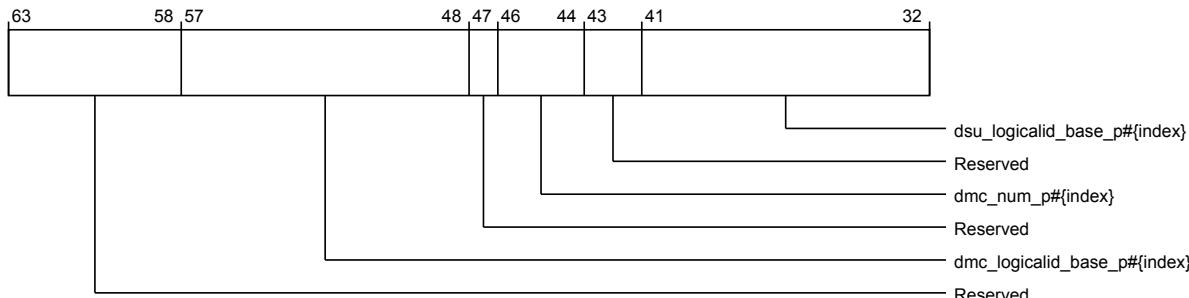


Figure 3-1065 por_mxp_por_mxp_p_0-5_info (high)

The following table shows the por_mxp_p_0-5_info higher register bit assignments.

Table 3-1085 por_mxp_por_mxp_p_0-5_info (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:48	dmc_logicalid_base_p#{index}	DMC AXIU interfaces logical ID base at this port (0 or 1)	RO	Configuration dependent
47	Reserved	Reserved	RO	-
46:44	dmc_num_p#{index}	Total number of SN-F AXIU interfaces at this port (0 to 4)	RO	Configuration dependent
43:42	Reserved	Reserved	RO	-
41:32	dsu_logicalid_base_p#{index}	DSU AXIU interfaces logical ID base at this port (0 or 1)	RO	Configuration dependent

The following image shows the lower register bit assignments.

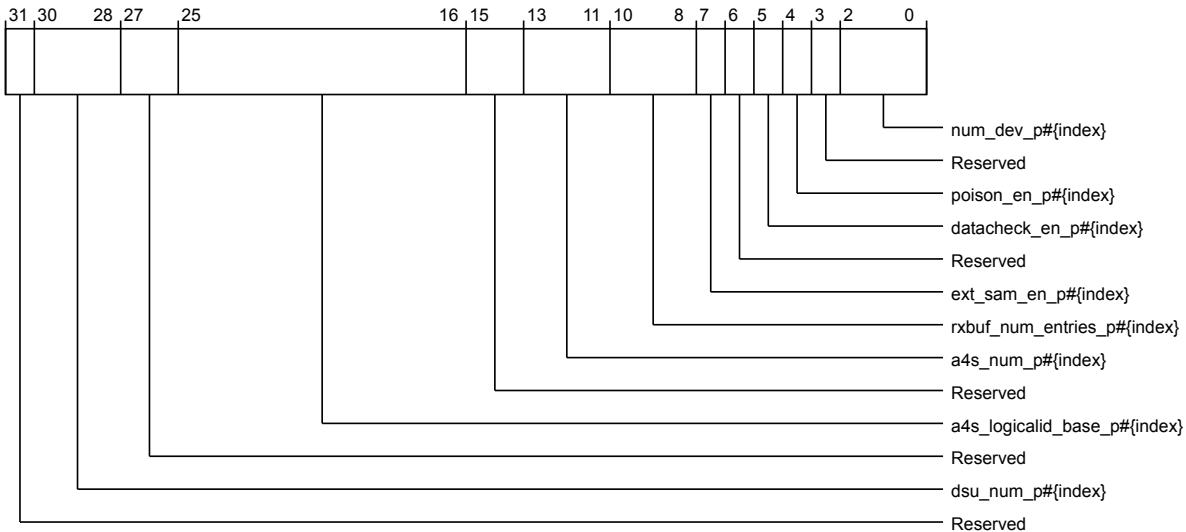


Figure 3-1066 por_mxp_por_mxp_p_0-5_info (low)

The following table shows the por_mxp_p_0-5_info lower register bit assignments.

Table 3-1086 por_mxp_por_mxp_p_0-5_info (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	dsu_num_p#{index}	Total number of RN-F AXIU interfaces at this port (0 to 4)	RO	Configuration dependent
27:26	Reserved	Reserved	RO	-
25:16	a4s_logicalid_base_p#{index}	AXI4Stream interfaces logical ID base at this port (0 or 1)	RO	Configuration dependent
15:14	Reserved	Reserved	RO	-
13:11	a4s_num_p#{index}	Total number of RN-F AXI4Stream interfaces at this port (0 to 4)	RO	Configuration dependent
10:8	rxbuf_num_entries_p#{index}	Number of input buffers at this port (2 to 4)	RO	Configuration dependent
7	ext_sam_en_p#{index}	ESAM enable	RO	Configuration dependent
6	Reserved	Reserved	RO	-
5	datacheck_en_p#{index}	Datacheck enable	RO	Configuration dependent
4	poison_en_p#{index}	Poison enable	RO	Configuration dependent
3	Reserved	Reserved	RO	-
2:0	num_dev_p#{index}	Number of devices connected to this port (0 to 4)	RO	Configuration dependent

por_mxp_p_0-5_info_1

This register repeats 5 times. It parametrized by the index from 0 to 5. Provides component identification information for XP port #{index}. NOTE: There will be max. of 6 MXP Port Info registers based on MXP_NUM_DEV_PORT_PARAM value. Each successive MXP Port Info register will be at the next 8

byte address boundary. Each successive MXP Port Info register will be named with the suffix. For example por_mxp_p<0:5>_info_1

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h908 + #{{16*[0, 1, 2, 3, 4, 5]}}
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

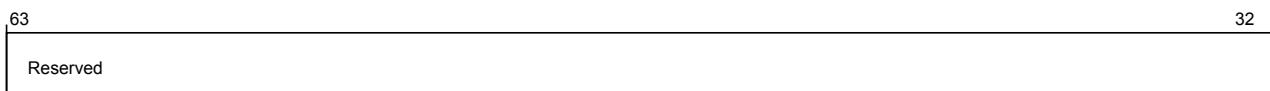


Figure 3-1067 por_mxp_por_mxp_p_0-5_info_1 (high)

The following table shows the por_mxp_p_0-5_info_1 higher register bit assignments.

Table 3-1087 por_mxp_por_mxp_p_0-5_info_1 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

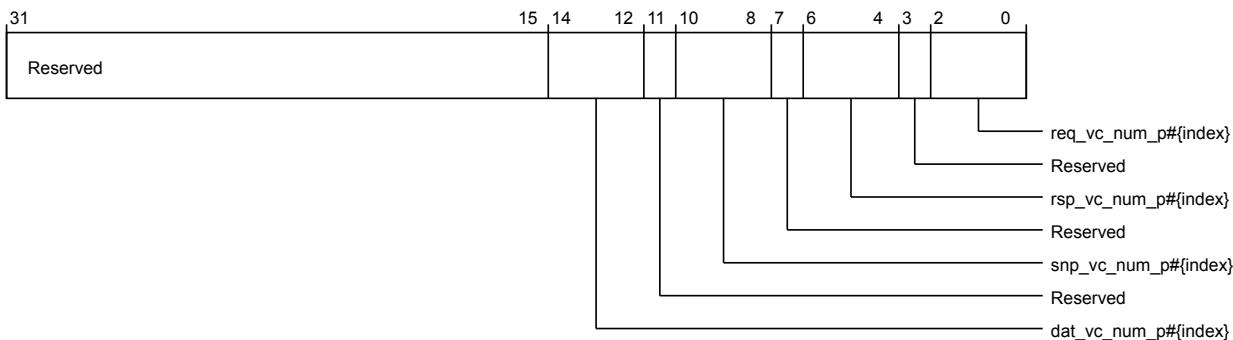


Figure 3-1068 por_mxp_por_mxp_p_0-5_info_1 (low)

The following table shows the por_mxp_p_0-5_info_1 lower register bit assignments.

Table 3-1088 por_mxp_por_mxp_p_0-5_info_1 (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	dat_vc_num_p#{index}	Number of replicated channels on DAT VC at this port	RO	Configuration dependent
11	Reserved	Reserved	RO	-

Table 3-1088 por_mxp_por_mxp_p_0-5_info_1 (low) (continued)

Bits	Field name	Description	Type	Reset
10:8	snp_vc_num_p#{index}	Number of replicated channels on SNP VC at this port	RO	Configuration dependent
7	Reserved	Reserved	RO	-
6:4	rsp_vc_num_p#{index}	Number of replicated channels on RSP VC at this port	RO	Configuration dependent
3	Reserved	Reserved	RO	-
2:0	req_vc_num_p#{index}	Number of replicated channels on REQ VC at this port	RO	Configuration dependent

por_dtm_unit_info

Provides component identification information for XP port 0 and 1.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h960
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1069 por_mxp_por_dtm_unit_info (high)

The following table shows the por_dtm_unit_info higher register bit assignments.

Table 3-1089 por_mxp_por_dtm_unit_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

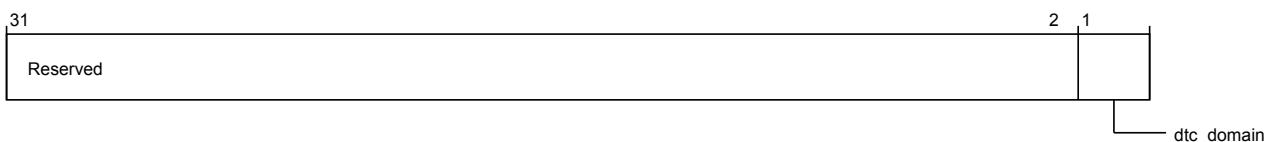


Figure 3-1070 por_mxp_por_dtm_unit_info (low)

The following table shows the por_dtm_unit_info lower register bit assignments.

Table 3-1090 por_mxp_por_dtm_unit_info (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1:0	dtc_domain	DTC domain number associated with this DTM	RO	Configuration dependent

por_dtm_unit_info_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Provides component identification information for XP ports # $\{2 * \text{index}\}$ and # $\{(2 * \text{index}) + 1\}$. NOTE: There will be max. of 3 DTM Unit Info registers based on MXP_MULTIPLE_DTM_EN_PARAM and MXP_NUM_DEV_PORT_PARAM value. Each successive DTM Unit Info register will be at the next 8 byte address boundary. Each successive DTM Unit Info register will be named with the suffix corresponding to the DT register number. For example por_dtm_unit_info_dt<1:3>

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h968 + # $\{8 * ([1, 2, 3] - 1)\}$
Register reset	Configuration dependent
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

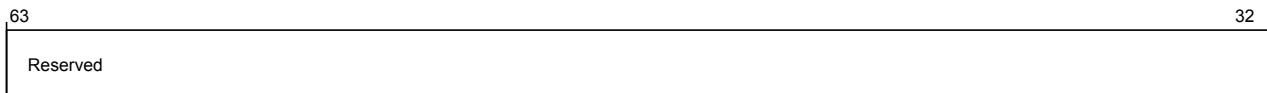


Figure 3-1071 por_mxp_por_dtm_unit_info_dt_1-3 (high)

The following table shows the por_dtm_unit_info_dt_1-3 higher register bit assignments.

Table 3-1091 por_mxp_por_dtm_unit_info_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

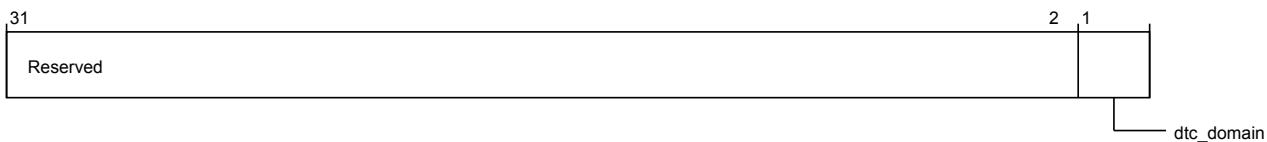


Figure 3-1072 por_mxp_por_dtm_unit_info_dt_1-3 (low)

The following table shows the por_dtm_unit_info_dt_1-3 lower register bit assignments.

Table 3-1092 por_mxp_por_dtm_unit_info_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1:0	dtc_domain	DTC domain number associated with this DTM	RO	Configuration dependent

por_mxp_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-1073 por_mxp_por_mxp_secure_register_groups_override (high)

The following table shows the por_mxp_secure_register_groups_override higher register bit assignments.

Table 3-1093 por_mxp_por_mxp_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

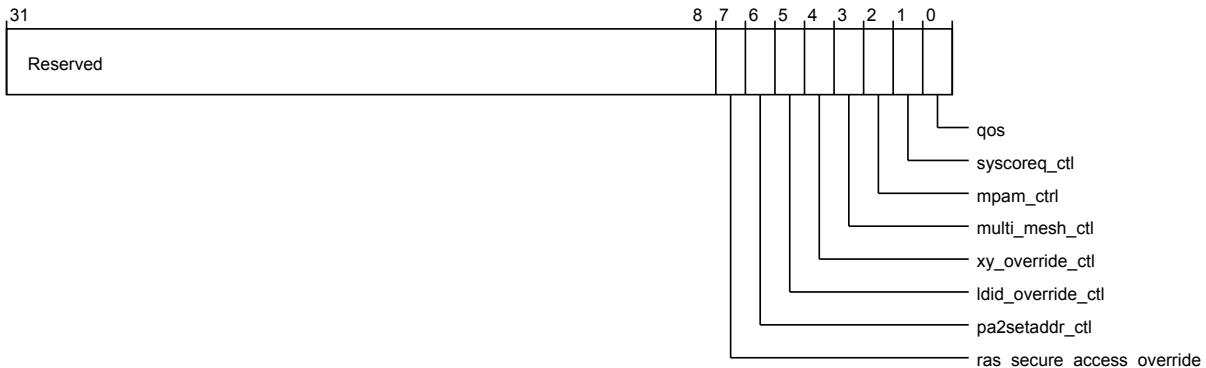


Figure 3-1074 por_mxp_por_mxp_secure_register_groups_override (low)

The following table shows the por_mxp_secure_register_groups_override lower register bit assignments.

Table 3-1094 por_mxp_por_mxp_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0
6	pa2setaddr_ctl	Allows non-secure access to secure PA to SETADDR control registers	RW	1'b0
5	ldid_override_ctl	Allows non-secure access to secure LDID override registers	RW	1'b0
4	xy_override_ctl	Allows non-secure access to secure XY override registers	RW	1'b0
3	multi_mesh_ctl	Allows non-secure access to secure Multi Mesh control registers	RW	1'b0
2	mpam_ctrl	Allows non-secure access to secure CHI port MPAM override register	RW	1'b0
1	syscoreq_ctl	Allows non-secure access to secure syscoreq_ctl registers	RW	1'b0
0	qos	Allows non-secure access to secure QoS registers	RW	1'b0

por_mxp_aux_ctl

Functions as the auxiliary control register for XP.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA00
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

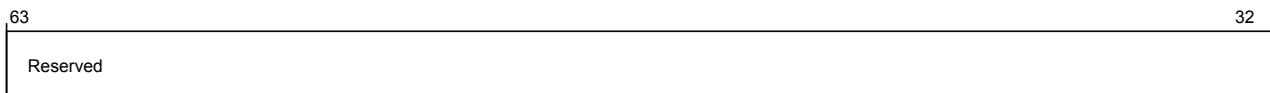


Figure 3-1075 por_mxp_por_mxp_aux_ctl (high)

The following table shows the por_mxp_aux_ctl higher register bit assignments.

Table 3-1095 por_mxp_por_mxp_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

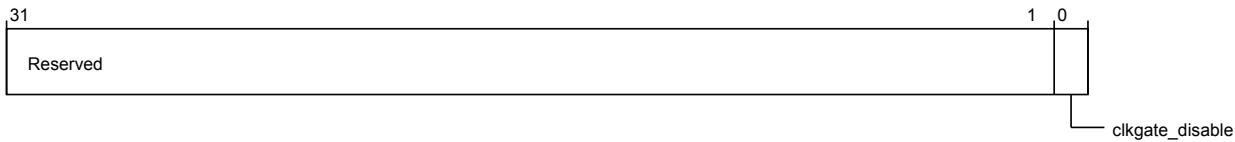


Figure 3-1076 por_mxp_por_mxp_aux_ctl (low)

The following table shows the por_mxp_aux_ctl lower register bit assignments.

Table 3-1096 por_mxp_por_mxp_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	clkgate_disable	Disables clock gating when set	RW	1'b0

por_mxp_device_port_ctl

Functions as the control register for XP device ports.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA08

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.

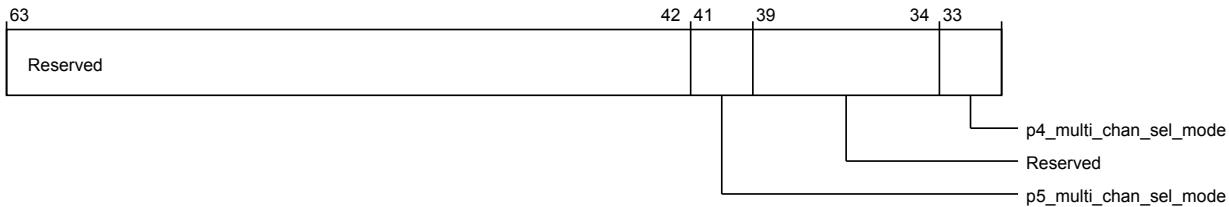


Figure 3-1077 por_mxp_por_mxp_device_port_ctl (high)

The following table shows the por_mxp_device_port_ctl higher register bit assignments.

Table 3-1097 por_mxp_por_mxp_device_port_ctl (high)

Bits	Field name	Description	Type	Reset
63:42	Reserved	Reserved	RO	-
41:40	p5_multi_chan_sel_mode	Selects the mode/scheme for channel selection in multi channel mesh structure 2'h0: Enable channel mapping based on TGTID scheme 2'h1: Enable channel mapping based on dynamic credit availability scheme 2'h2: Enable channel mapping based on direct connect scheme 2'h3: Reserved	RW	2'b0
39:34	Reserved	Reserved	RO	-
33:32	p4_multi_chan_sel_mode	Selects the mode/scheme for channel selection in multi channel mesh structure 2'h0: Enable channel mapping based on TGTID scheme 2'h1: Enable channel mapping based on dynamic credit availability scheme 2'h2: Enable channel mapping based on direct connect scheme 2'h3: Reserved	RW	2'b0

The following image shows the lower register bit assignments.

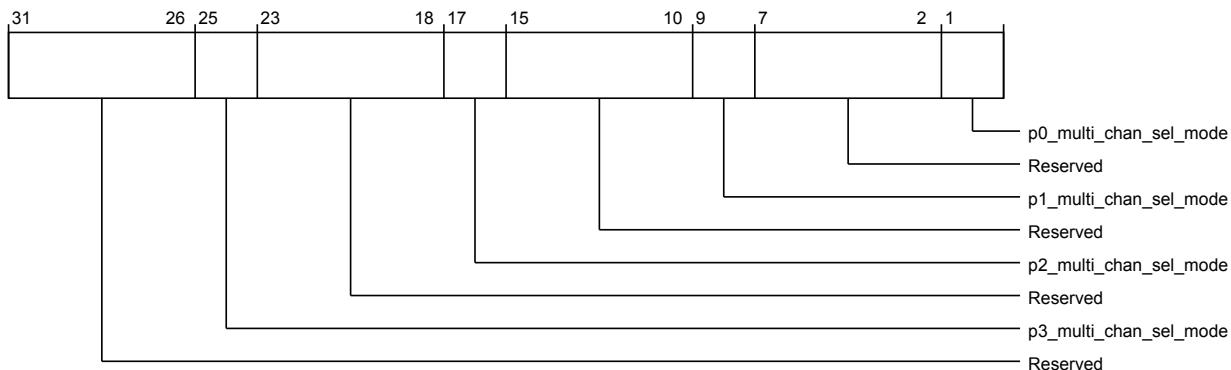


Figure 3-1078 por_mxp_por_mxp_device_port_ctl (low)

The following table shows the por_mxp_device_port_ctl lower register bit assignments.

Table 3-1098 por_mxp_por_mxp_device_port_ctl (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	p3_multi_chan_sel_mode	Selects the mode/scheme for channel selection in multi channel mesh structure 2'h0: Enable channel mapping based on TGTID scheme 2'h1: Enable channel mapping based on dynamic credit availability scheme 2'h2: Enable channel mapping based on direct connect scheme 2'h3: Reserved	RW	2'b0
23:18	Reserved	Reserved	RO	-

Table 3-1098 por_mxp_por_mxp_device_port_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
17:16	p2_multi_chan_sel_mode	Selects the mode/scheme for channel selection in multi channel mesh structure 2'h0: Enable channel mapping based on TGTID scheme 2'h1: Enable channel mapping based on dynamic credit availability scheme 2'h2: Enable channel mapping based on direct connect scheme 2'h3: Reserved	RW	2'b0
15:10	Reserved	Reserved	RO	-
9:8	p1_multi_chan_sel_mode	Selects the mode/scheme for channel selection in multi channel mesh structure 2'h0: Enable channel mapping based on TGTID scheme 2'h1: Enable channel mapping based on dynamic credit availability scheme 2'h2: Enable channel mapping based on direct connect scheme 2'h3: Reserved	RW	2'b0
7:2	Reserved	Reserved	RO	-
1:0	p0_multi_chan_sel_mode	Selects the mode/scheme for channel selection in multi channel mesh structure 2'h0: Enable channel mapping based on TGTID scheme 2'h1: Enable channel mapping based on dynamic credit availability scheme 2'h2: Enable channel mapping based on direct connect scheme 2'h3: Reserved	RW	2'b0

por_mxp_p_0-5_mpam_override

This register repeats 5 times. It parametrized by the index from 0 to 5. Controls MPAM fields for devices connected to port # {index}. Valid only if the devices doesn't support MPAM.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA10 + # {8*[0, 1, 2, 3, 4, 5]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mxp_secure_register_groups_override.mpam_ctrl

The following image shows the higher register bit assignments.

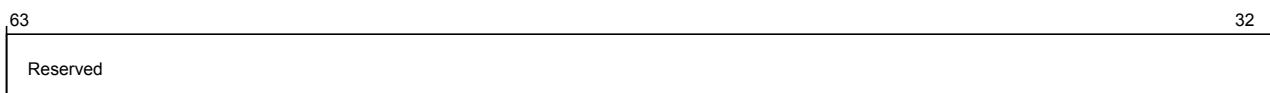


Figure 3-1079 por_mxp_por_mxp_p_0-5_mpam_override (high)

The following table shows the por_mxp_p_0-5_mpam_override higher register bit assignments.

Table 3-1099 por_mxp_por_mxp_p_0-5_mpam_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

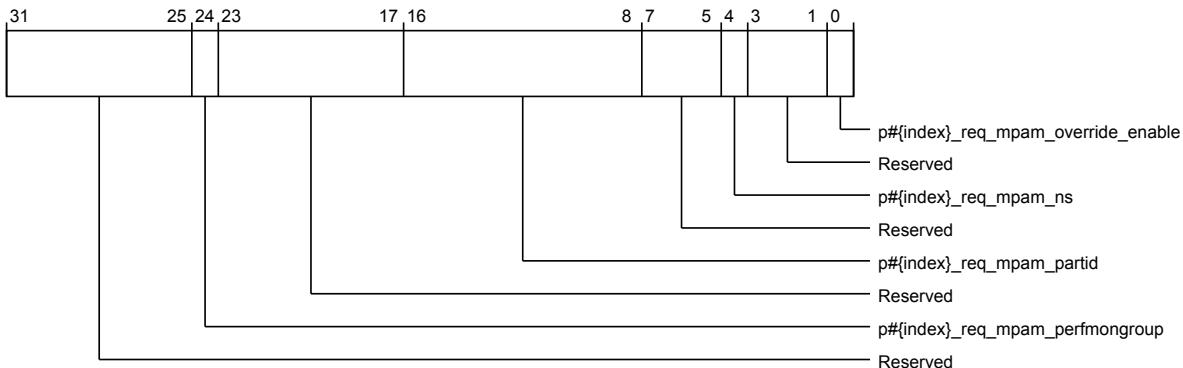


Figure 3-1080 por_mxp_por_mxp_p_0-5_mpam_override (low)

The following table shows the por_mxp_p_0-5_mpam_override lower register bit assignments.

Table 3-1100 por_mxp_por_mxp_p_0-5_mpam_override (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	p#{index}_req_mpam_perfmongroup	MPAM.PerfMonGroup sub-field that overrides the REQ channel MPAM.PerfMonGroup when p#{index}_req_mpam_override_enable is set	RW	1'b0
23:17	Reserved	Reserved	RO	-
16:8	p#{index}_req_mpam_partid	MPAM.PartID sub-field that overrides the REQ channel MPAM.PartID when p#{index}_req_mpam_override_enable is set	RW	9'b0
7:5	Reserved	Reserved	RO	-
4	p#{index}_req_mpam_ns	MPAM.NS sub-field that overrides the REQ channel MPAM.NS when p#{index}_req_mpam_override_enable is set	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	p#{index}_req_mpam_override_enable	P#{index} DEV MPAM Override Enable on REQ Channel: 1 - Drive the MPAM fields on REQ channel with the values from this register, 0 - Override of MPAM fields in REQ channel is disabled	RW	1'b0

por_mxp_p_0-5_ldid_override

This register repeats 5 times. It parametrized by the index from 0 to 5. Controls LDID fields in REQ FLIT for devices connected to port #{index}. Valid only if POR_MXP_RNF_CLUSTER_EN_PARAM is 1.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA40 + #{8*[0, 1, 2, 3, 4, 5]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mxp_secure_register_groups_override.ldid_override_ctl

The following image shows the higher register bit assignments.

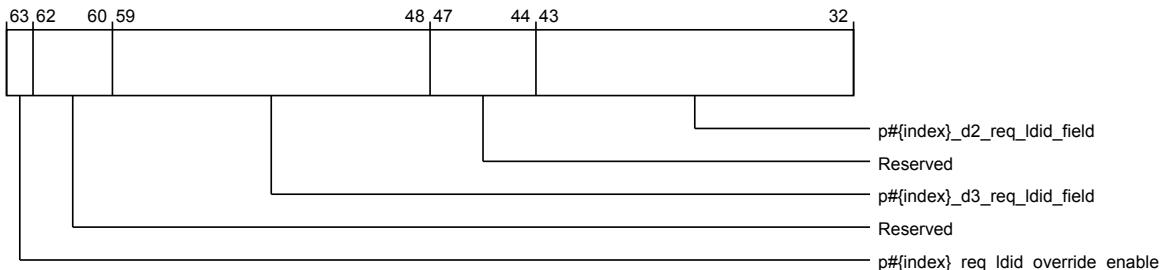


Figure 3-1081 por_mxp_por_mxp_p_0-5_ldid_override (high)

The following table shows the por_mxp_p_0-5_ldid_override higher register bit assignments.

Table 3-1101 por_mxp_por_mxp_p_0-5_ldid_override (high)

Bits	Field name	Description	Type	Reset
63	p#{index}_req_ldid_override_enable	P#{index} DEV LDID Override Enable on REQ Channel: 1 - Drive the LDID fields on REQ channel with the values from this register, 0 - Override of LDID fields in REQ channel is disabled	RW	1'b0
62:60	Reserved	Reserved	RO	-
59:48	p#{index}_d3_req_ldid_field	LDID value that overrides the P#{index}_D3 REQ channel LDID field when p#{index}_req_ldid_override_enable is set	RW	12'b0
47:44	Reserved	Reserved	RO	-
43:32	p#{index}_d2_req_ldid_field	LDID value that overrides the P#{index}_D2 REQ channel LDID field when p#{index}_req_ldid_override_enable is set	RW	12'b0

The following image shows the lower register bit assignments.

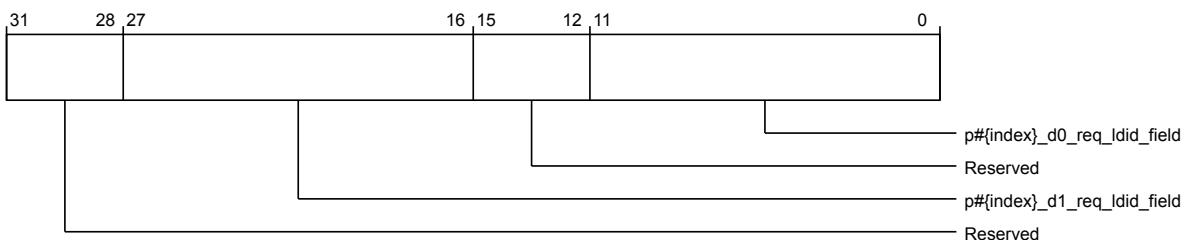


Figure 3-1082 por_mxp_por_mxp_p_0-5_ldid_override (low)

The following table shows the por_mxp_p_0-5_ldid_override lower register bit assignments.

Table 3-1102 por_mxp_por_mxp_p_0-5_ldid_override (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:16	p#{index}_d1_req_ldid_field	LDID value that overrides the P#{index}_D1 REQ channel LDID field when p#{index}_req_ldid_override_enable is set	RW	12'b0
15:12	Reserved	Reserved	RO	-
11:0	p#{index}_d0_req_ldid_field	LDID value that overrides the P#{index}_D0 REQ channel LDID field when p#{index}_req_ldid_override_enable is set	RW	12'b0

por_mxp_p_0-5_qos_control

This register repeats 5 times. It parametrized by the index from 0 to 5. Controls QoS settings for devices connected to port #{index}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA80 + #{32*[0, 1, 2, 3, 4, 5]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mxp_secure_register_groups_override.qos

The following image shows the higher register bit assignments.



Figure 3-1083 por_mxp_por_mxp_p_0-5_qos_control (high)

The following table shows the por_mxp_p_0-5_qos_control higher register bit assignments.

Table 3-1103 por_mxp_por_mxp_p_0-5_qos_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

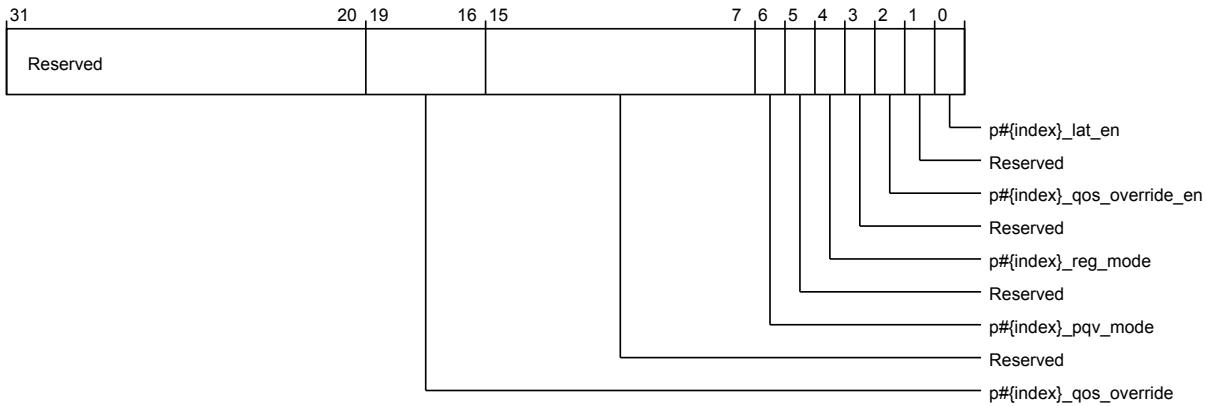


Figure 3-1084 por_mxp_por_mxp_p_0-5_qos_control (low)

The following table shows the por_mxp_p_0-5_qos_control lower register bit assignments.

Table 3-1104 por_mxp_por_mxp_p_0-5_qos_control (low)

Bits	Field name	Description	Type	Reset
31:20	Reserved	Reserved	RO	-
19:16	p#{index}_qos_override	QoS override value for port #{index}	RW	4'b0000
15:7	Reserved	Reserved	RO	-
6	p#{index}_pqv_mode	Configures the QoS regulator mode during period mode 1'b0: Normal mode; QoS value is stable when the master is idle 1'b1: Quiesce high mode; QoS value tends to the maximum value when the master is idle	RW	1'b0
5	Reserved	Reserved	RO	-
4	p#{index}_reg_mode	Configures the QoS regulator mode 1'b0: Latency mode 1'b1: Period mode; used for bandwidth regulation	RW	1'b0
3	Reserved	Reserved	RO	-
2	p#{index}_qos_override_en	Enables port #{index} QoS override; when set, allows QoS value on inbound transactions to be overridden	RW	1'b0
1	Reserved	Reserved	RO	-
0	p#{index}_lat_en	Enables port #{index} QoS regulation when set	RW	1'b0

por_mxp_p_0-5_qos_lat_tgt

This register repeats 5 times. It parametrized by the index from 0 to 5. Controls QoS target latency/period (in cycles) for regulation of devices connected to port #{index}.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA88 + #{32*[0, 1, 2, 3, 4, 5]}

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_mxp_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

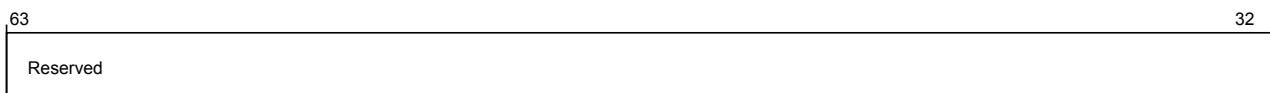


Figure 3-1085 por_mxp_por_mxp_p_0-5_qos_lat_tgt (high)

The following table shows the por_mxp_p_0-5_qos_lat_tgt higher register bit assignments.

Table 3-1105 por_mxp_por_mxp_p_0-5_qos_lat_tgt (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

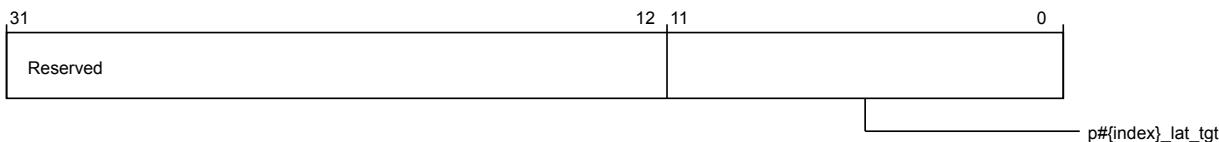


Figure 3-1086 por_mxp_por_mxp_p_0-5_qos_lat_tgt (low)

The following table shows the por_mxp_p_0-5_qos_lat_tgt lower register bit assignments.

Table 3-1106 por_mxp_por_mxp_p_0-5_qos_lat_tgt (low)

Bits	Field name	Description	Type	Reset
31:12	Reserved	Reserved	RO	-
11:0	p#{index}_lat_tgt	Port #{index} transaction target latency/period; a value of 0 corresponds to no regulation	RW	12'h000

por_mxp_p_0-5_qos_lat_scale

This register repeats 5 times. It parametrized by the index from 0 to 5. Controls the QoS target scale factor for devices connected to port #{index}. The scale factor is represented in powers of two from the range 2^{-3} to 2^{-10} .

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA90 + #{32*[0, 1, 2, 3, 4, 5]}

Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mxp_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

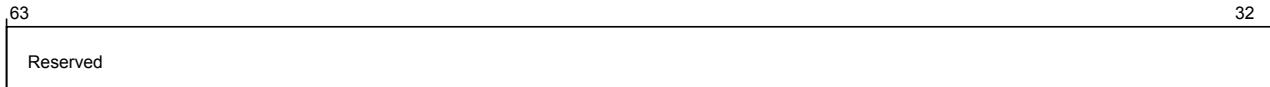


Figure 3-1087 por_mxp_por_mxp_p_0-5_qos_lat_scale (high)

The following table shows the por_mxp_p_0-5_qos_lat_scale higher register bit assignments.

Table 3-1107 por_mxp_por_mxp_p_0-5_qos_lat_scale (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

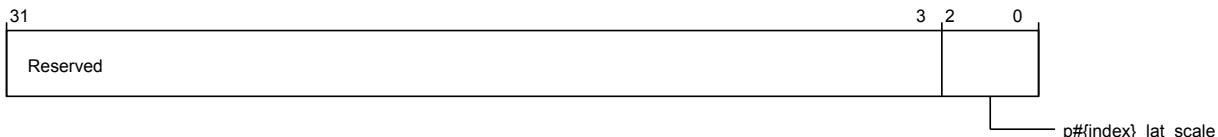


Figure 3-1088 por_mxp_por_mxp_p_0-5_qos_lat_scale (low)

The following table shows the por_mxp_p_0-5_qos_lat_scale lower register bit assignments.

Table 3-1108 por_mxp_por_mxp_p_0-5_qos_lat_scale (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2:0	p#{index}_lat_scale	Port 0 QoS scale factor 3'b000: 2 ⁽⁻³⁾ 3'b001: 2 ⁽⁻⁴⁾ 3'b010: 2 ⁽⁻⁵⁾ 3'b011: 2 ⁽⁻⁶⁾ 3'b100: 2 ⁽⁻⁷⁾ 3'b101: 2 ⁽⁻⁸⁾ 3'b110: 2 ⁽⁻⁹⁾ 3'b111: 2 ⁽⁻¹⁰⁾	RW	3'h0

por_mxp_p_0-5_qos_lat_range

This register repeats 5 times. It parametrized by the index from 0 to 5. Controls the minimum and maximum QoS values generated by the QoS regulator for devices connected to port # $\{index\}$.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA98 + # $\{32*[0, 1, 2, 3, 4, 5]\}$
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group	por_mxp_secure_register_groups_override.qos_override

The following image shows the higher register bit assignments.



Figure 3-1089 por_mxp_por_mxp_p_0-5_qos_lat_range (high)

The following table shows the por_mxp_p_0-5_qos_lat_range higher register bit assignments.

Table 3-1109 por_mxp_por_mxp_p_0-5_qos_lat_range (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

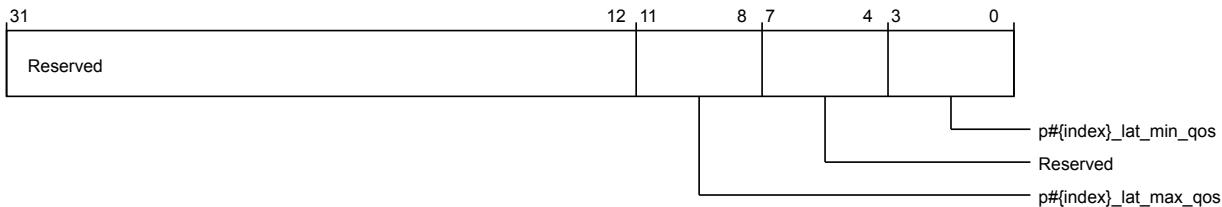


Figure 3-1090 por_mxp_por_mxp_p_0-5_qos_lat_range (low)

The following table shows the por_mxp_p_0-5_qos_lat_range lower register bit assignments.

Table 3-1110 por_mxp_por_mxp_p_0-5_qos_lat_range (low)

Bits	Field name	Description	Type	Reset
31:12	Reserved	Reserved	RO	-
11:8	p#{index}_lat_max_qos	Port # $\{index\}$ QoS maximum value	RW	4'h0

Table 3-1110 por_mxp_por_mxp_p_0-5_qos_lat_range (low) (continued)

Bits	Field name	Description	Type	Reset
7:4	Reserved	Reserved	RO	-
3:0	p#{index}_lat_min_qos	Port #{index} QoS minimum value	RW	4'h0

por_mxp_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2

Register reset 64'b0

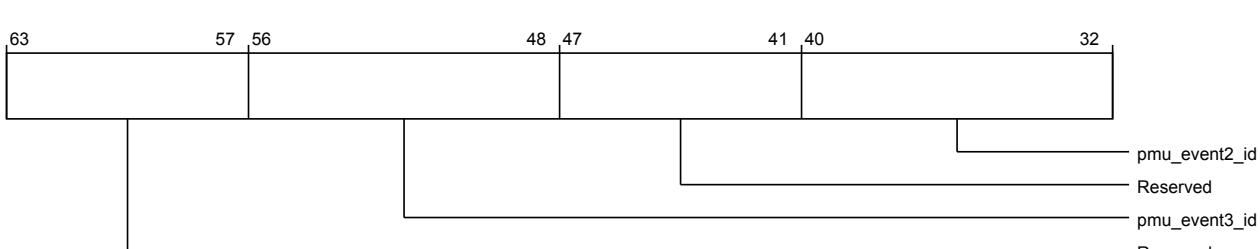


Figure 3-1091 por mxp nor mxp pmu event sel (high)

The following table shows the por_mxp_ppmu_event_sel higher register bit assignments

Table 3-1111 por mxp por mxp pmu event sel (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56:48	pmu_event3_id	XP PMU Event 3 ID; see pmu_event0_id for encodings	RW	9'b0
47:41	Reserved	Reserved	RO	-
40:32	pmu_event2_id	XP PMU Event 2 ID; see pmu_event0_id for encodings	RW	9'b0

The following image shows the lower register bit assignments.

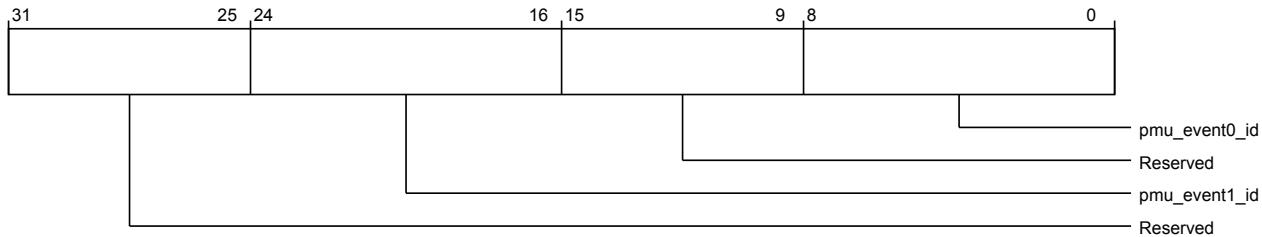


Figure 3-1092 por_mxp_por_mxp_pmu_event_sel (low)

The following table shows the por_mxp_pmu_event_sel lower register bit assignments.

Table 3-1112 por_mxp_por_mxp_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24:16	pmu_event1_id	XP PMU Event 1 ID; see pmu_event0_id for encodings	RW	9'b0

Table 3-1112 por_mxp_por_mxp_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
15:9	Reserved	Reserved	RO	-
8:0	pmu_event0_id	XP PMU Event 0 ID Bits [8:5]: PC 4'b0000: REQ; REQ channel when POR_REQ_VC_NUM_PARAM = 1 ; REQ Sub-channel 1: when POR_REQ_VC_NUM_PARAM > 1 4'b0001: RSP; RSP channel when POR_RSP_VC_NUM_PARAM = 1 ; RSP Sub-channel 1: when POR_RSP_VC_NUM_PARAM > 1 4'b0010: SNP; SNP channel when POR_SNP_VC_NUM_PARAM = 1 ; SNP Sub-channel 1: when POR_SNP_VC_NUM_PARAM > 1 4'b0011: DAT; DAT channel when POR_DAT_VC_NUM_PARAM = 1 ; DAT Sub-channel 1: when POR_DAT_VC_NUM_PARAM > 1 4'b0100: PUB 4'b0101: RSP2; RSP Sub-channel 2: Applicable when POR_RSP_VC_NUM_PARAM > 1 4'b0110: DAT2; DAT Sub-channel 2: Applicable when POR_DAT_VC_NUM_PARAM > 1 4'b0111: REQ2; REQ Sub-channel 2: Applicable when POR_REQ_VC_NUM_PARAM > 1 4'b1000: SNP2; SNP Sub-channel 2: Applicable when POR_SNP_VC_NUM_PARAM > 1 Bits [4:2]: Interface 3'b000: East when NUM_XP > 1 ; Device port 0 when NUM_XP == 1 (Single XP config) 3'b001: West when NUM_XP > 1 ; Device port 1 when NUM_XP == 1 (Single XP config) 3'b010: North when NUM_XP > 1 ; Device port 2 when NUM_XP == 1 (Single XP config) 3'b011: South when NUM_XP > 1 ; Device port 3 when NUM_XP == 1 (Single XP config) 3'b100: Device port 0 when NUM_XP > 1 ; Device port 4 when NUM_XP == 1 (Single XP config) 3'b101: Device port 1 when NUM_XP > 1 ; Device port 5 when NUM_XP == 1 (Single XP config) 3'b110: Device port 2 when NUM_XP > 1 ; No Selection when NUM_XP == 1 (Single XP config) 3'b111: Device port 3 when NUM_XP > 1 ; No Selection when NUM_XP == 1 (Single XP config) Bits [1:0]: Event specifier 2'b00: No event 2'b01: TX flit valid; signaled when a flit is successfully transmitted 2'b10: TX flit stall; signaled when flit transmission is stalled and waiting on credits 2'b11: Partial DAT flit; signaled when 128-bit DAT flits could not be merged into a 256-bit DAT flit; only applicable on the DAT PC on RN-F CHIA and RN-F CHIA ESAM ports	RW	9'b0

por_mxp_errfr

Functions as the error feature register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3000
Register reset	64'b0000010100101
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mxp_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-1093 por_mxp_por_mxp_errfr (high)

The following table shows the por_mxp_errfr higher register bit assignments.

Table 3-1113 por_mxp_por_mxp_errfr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

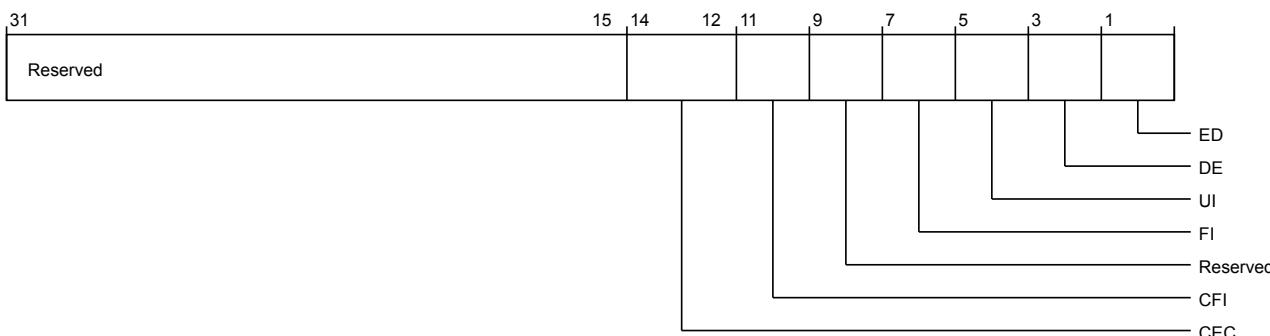


Figure 3-1094 por_mxp_por_mxp_errfr (low)

The following table shows the por_mxp_errfr lower register bit assignments.

Table 3-1114 por_mxp_por_mxp_errfr (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00

Table 3-1114 por_mxp_por_mxp_errfr (low) (continued)

Bits	Field name	Description	Type	Reset
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_mxp_errctlr

Functions as the error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3008
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group	por_mxp_secure_register_groups_override.ras_secure_access_override
override	

The following image shows the higher register bit assignments.

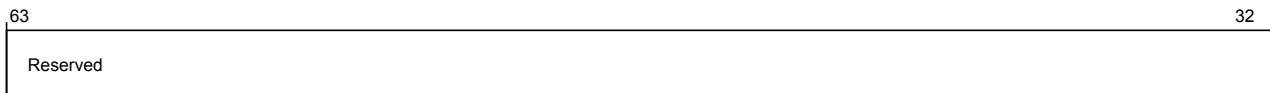


Figure 3-1095 por_mxp_por_mxp_errctlr (high)

The following table shows the por_mxp_errctlr higher register bit assignments.

Table 3-1115 por_mxp_por_mxp_errctlr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

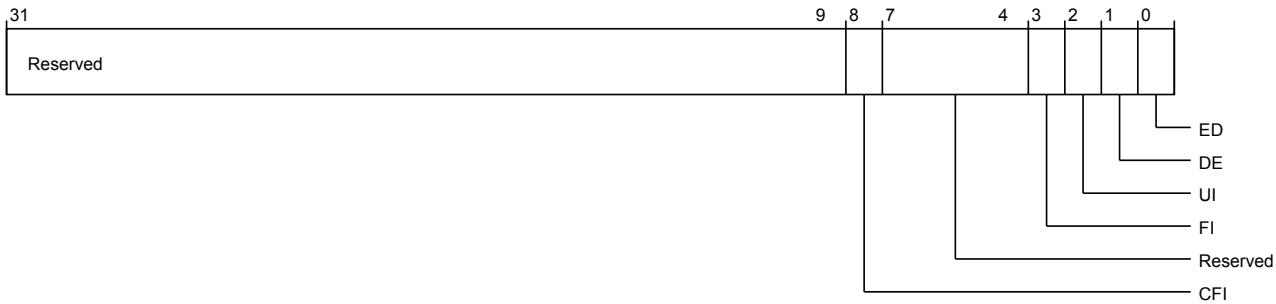


Figure 3-1096 por_mxp_por_mxp_errctlr (low)

The following table shows the por_mxp_errctlr lower register bit assignments.

Table 3-1116 por_mxp_por_mxp_errctlr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_mxp_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_mxp_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_mxp_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_mxp_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_mxp_errfr.ED	RW	1'b0

por_mxp_errstatus

Functions as the error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3010
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mxp_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

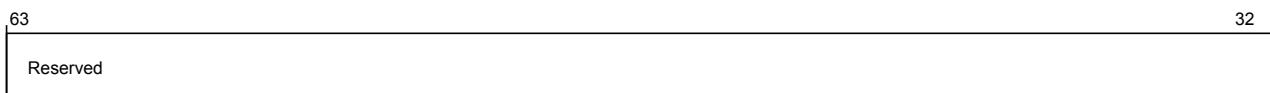


Figure 3-1097 por_mxp_por_mxp_errstatus (high)

The following table shows the por_mxp_errstatus higher register bit assignments.

Table 3-1117 por_mxp_por_mxp_errstatus (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-1098 por_mxp_por_mxp_errstatus (low)

The following table shows the por_mxp_errstatus lower register bit assignments.

Table 3-1118 por_mxp_por_mxp_errstatus (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0

Table 3-1118 por_mxp_por_mxp_errstatus (low) (continued)

Bits	Field name	Description	Type	Reset
26	MV	por_mxp_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_mxp_errmisc

Functions as the miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3028
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mxp_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

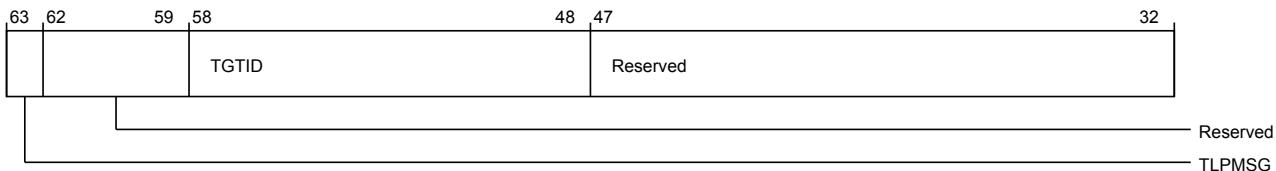


Figure 3-1099 por_mxp_por_mxp_errmisc (high)

The following table shows the por_mxp_errmisc higher register bit assignments.

Table 3-1119 por_mxp_por_mxp_errmisc (high)

Bits	Field name	Description	Type	Reset
63	TLPMSG	Error flit TLPMSG Status	RW	1'b0
62:59	Reserved	Reserved	RO	-
58:48	TGTID	Error flit target ID	RW	11'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

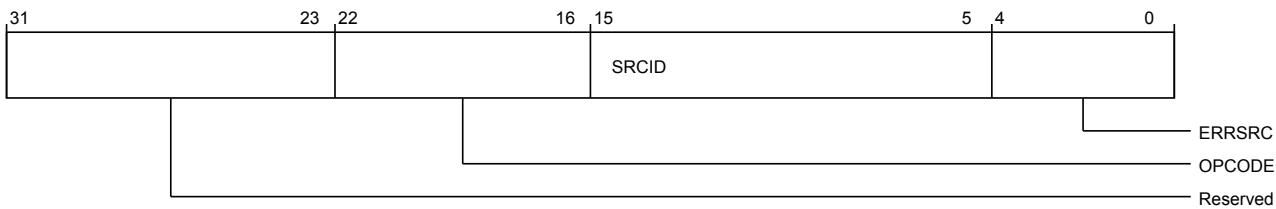


Figure 3-1100 por_mxp_por_mxp_errmisc (low)

The following table shows the por_mxp_errmisc lower register bit assignments.

Table 3-1120 por_mxp_por_mxp_errmisc (low)

Bits	Field name	Description	Type	Reset
31:23	Reserved	Reserved	RO	-
22:16	OPCODE	Error flit opcode	RW	7'b0

Table 3-1120 por_mxp_por_mxp_errmisc (low) (continued)

Bits	Field name	Description	Type	Reset
15:5	SRCID	Error flit source ID	RW	11'b0

Table 3-1120 por_mxp_por_mxp_errmisc (low) (continued)

Bits	Field name	Description	Type	Reset
4:0	ERRSRC	<p>Error source for Mesh With Replicated Channels:</p> <p>Bits [4:2]: Transaction type</p> <p>3'b000: REQ</p> <p>3'b001: RSP</p> <p>3'b010: SNP</p> <p>3'b011: DAT</p> <p>3'b100: REQ2</p> <p>3'b101: RSP2</p> <p>3'b110: SNP2</p> <p>3'b111: DAT2</p> <p>Bits [1:0]: Port</p> <p>2'b00: Port 0</p> <p>2'b01: Port 1</p> <p>2'b10: Port 2</p> <p>2'b11: Port 3</p> <p>Mesh Without Replicated Channels:</p> <p>Bits [4:2]: Transaction type</p> <p>3'b000: REQ</p> <p>3'b001: RSP</p> <p>3'b010: SNP</p> <p>3'b011: DAT</p> <p>3'b100: Reserved</p> <p>3'b101: Reserved</p> <p>3'b110: Reserved</p> <p>3'b111: Reserved</p> <p>Bits [1:0]: Port</p> <p>2'b00: Port 0</p> <p>2'b01: Port 1</p> <p>2'b10: Port 2</p> <p>2'b11: Port 3</p> <p>Mesh With Single MXP (1x1 Mesh):</p> <p>Bits [4:3]: Transaction type</p> <p>2'b00: REQ</p> <p>2'b01: RSP</p> <p>2'b10: SNP</p>	RW	5'b0

por_mxp_p_0-5_byte_par_err_inj

This register repeats 5 times. It parametrized by the index from 0 to 5. Functions as the byte parity error injection register for XP port # $\{\text{index}\}$.

Its characteristics are:

Type	WO
Register width (Bits)	64
Address offset	16'h3030 + # $\{8*[0, 1, 2, 3, 4, 5]\}$
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-1101 por_mxp_p_0-5_byte_par_err_inj (high)

The following table shows the por_mxp_p_0-5_byte_par_err_inj higher register bit assignments.

Table 3-1121 por_mxp_p_0-5_byte_par_err_inj (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

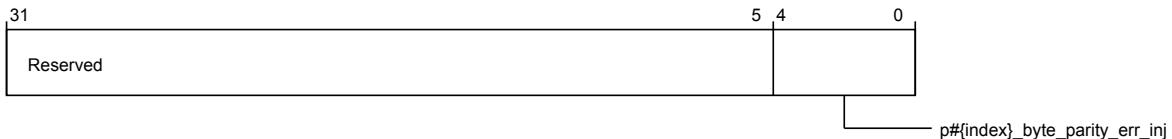


Figure 3-1102 por_mxp_p_0-5_byte_par_err_inj (low)

The following table shows the por_mxp_p_0-5_byte_par_err_inj lower register bit assignments.

Table 3-1122 por_mxp_p_0-5_byte_par_err_inj (low)

Bits	Field name	Description	Type	Reset
31:5	Reserved	Reserved	RO	-
4:0	p#\{index\}_byte_parity_err_inj	Specifies a byte lane; once this register is written, a byte parity error is injected in the specified byte lane on the next DAT flit upload NOTE: Only applicable if an RN-F is attached to port # $\{\text{index}\}$. Byte parity error is only injected if the RN-F is configured to not support Datacheck.	WO	5'h00

por_mxp_errfr_NS

Functions as the non-secure error feature register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3100
Register reset	64'b0000010100101
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1103 por_mxp_por_mxp_errfr_ns (high)

The following table shows the por_mxp_errfr_NS higher register bit assignments.

Table 3-1123 por_mxp_por_mxp_errfr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

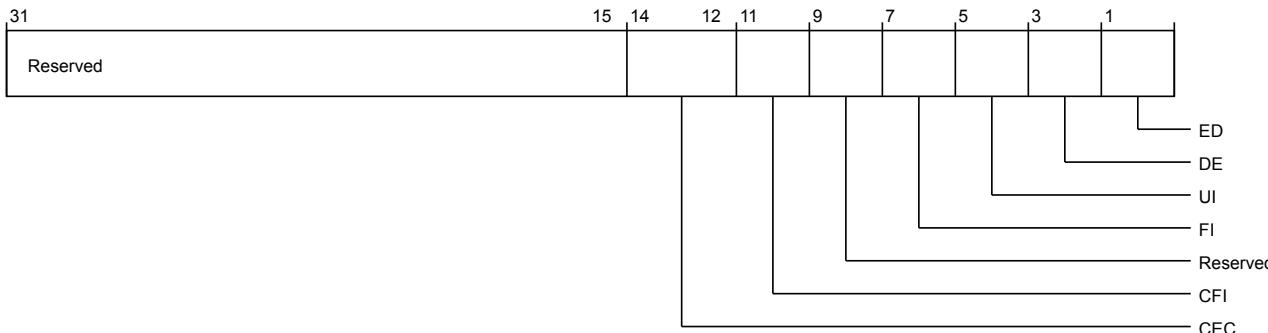


Figure 3-1104 por_mxp_por_mxp_errfr_ns (low)

The following table shows the por_mxp_errfr_NS lower register bit assignments.

Table 3-1124 por_mxp_por_mxp_errfr_ns (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model	RO	3'b000
11:10	CFI	Corrected error interrupt	RO	2'b00
9:8	Reserved	Reserved	RO	-

Table 3-1124 por_mxp_por_mxp_errfr_ns (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_mxp_errctlr_NS

Functions as the non-secure error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3108

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1105 por_mxp_por_mxp_errctlr_ns (high)

The following table shows the por_mxp_errctlr_NS higher register bit assignments.

Table 3-1125 por_mxp_por_mxp_errctlr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

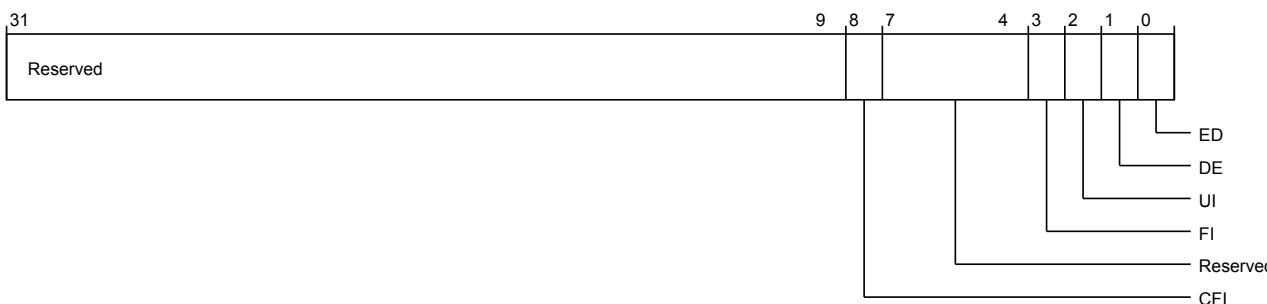


Figure 3-1106 por_mxp_por_mxp_errctlr_ns (low)

The following table shows the por_mxp_errctlr_NS lower register bit assignments.

Table 3-1126 por_mxp_por_mxp_errctlr_ns (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_mxp_errfr_NS.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_mxp_errfr_NS.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_mxp_errfr_NS.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_mxp_errfr_NS.DE	RW	1'b0
0	ED	Enables error detection as specified in por_mxp_errfr_NS.ED	RW	1'b0

por_mxp_errstatus_NS

Functions as the non-secure error status register.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h3110

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

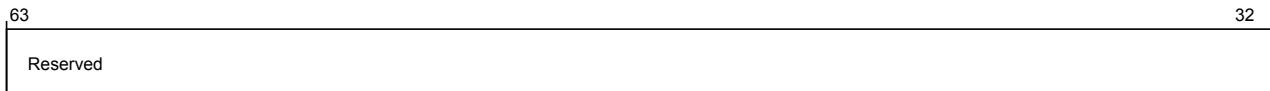


Figure 3-1107 por_mxp_por_mxp_errstatus_ns (high)

The following table shows the por_mxp_errstatus_NS higher register bit assignments.

Table 3-1127 por_mxp_por_mxp_errstatus_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-1108 por_mxp_por_mxp_errstatus_ns (low)

The following table shows the por_mxp_errstatus_NS lower register bit assignments.

Table 3-1128 por_mxp_por_mxp_errstatus_ns (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_mxp_errmisc_NS valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-

Table 3-1128 por_mxp_por_mxp_errstatus_ns (low) (continued)

Bits	Field name	Description	Type	Reset
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_mxp_errmisc_NS

Functions as the non-secure miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3128

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

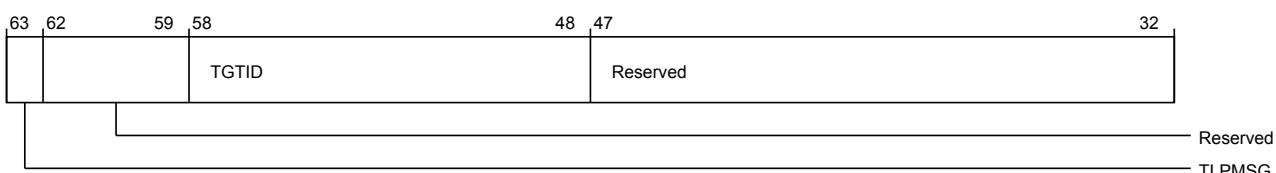


Figure 3-1109 por_mxp_por_mxp_errmisc_ns (high)

The following table shows the por_mxp_errmisc_NS higher register bit assignments.

Table 3-1129 por_mxp_por_mxp_errmisc_ns (high)

Bits	Field name	Description	Type	Reset
63	TLPMSG	Error flit TLPMSG Status	RW	1'b0
62:59	Reserved	Reserved	RO	-
58:48	TGTID	Error flit target ID	RW	11'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

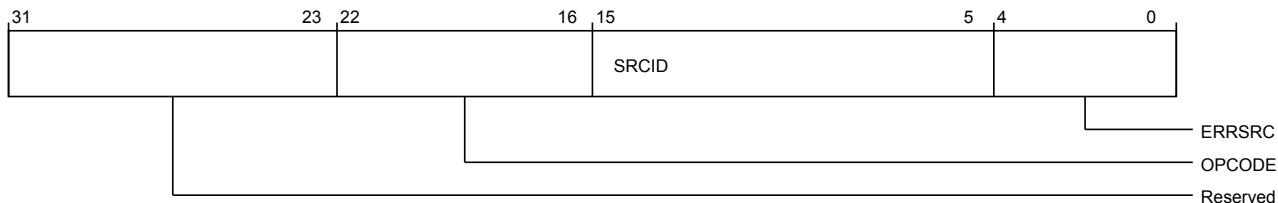


Figure 3-1110 por_mxp_por_mxp_errmisc_ns (low)

The following table shows the por_mxp_errmisc_NS lower register bit assignments.

Table 3-1130 por_mxp_por_mxp_errmisc_ns (low)

Bits	Field name	Description	Type	Reset
31:23	Reserved	Reserved	RO	-
22:16	OPCODE	Error flit opcode	RW	7'b0

Table 3-1130 por_mxp_por_mxp_errmisc_ns (low) (continued)

Bits	Field name	Description	Type	Reset
15:5	SRCID	Error flit source ID	RW	11'b0

Table 3-1130 por_mxp_por_mxp_errmisc_ns (low) (continued)

Bits	Field name	Description	Type	Reset
4:0	ERRSRC	<p>Error source for Mesh With Replicated Channels:</p> <p>Bits [4:2]: Transaction type</p> <p>3'b000: REQ</p> <p>3'b001: RSP</p> <p>3'b010: SNP</p> <p>3'b011: DAT</p> <p>3'b100: REQ2</p> <p>3'b101: RSP2</p> <p>3'b110: SNP2</p> <p>3'b111: DAT2</p> <p>Bits [1:0]: Port</p> <p>2'b00: Port 0</p> <p>2'b01: Port 1</p> <p>2'b10: Port 2</p> <p>2'b11: Port 3</p> <p>Mesh Without Replicated Channels:</p> <p>Bits [4:2]: Transaction type</p> <p>3'b000: REQ</p> <p>3'b001: RSP</p> <p>3'b010: SNP</p> <p>3'b011: DAT</p> <p>3'b100: Reserved</p> <p>3'b101: Reserved</p> <p>3'b110: Reserved</p> <p>3'b111: Reserved</p> <p>Bits [1:0]: Port</p> <p>2'b00: Port 0</p> <p>2'b01: Port 1</p> <p>2'b10: Port 2</p> <p>2'b11: Port 3</p> <p>Mesh With Single MXP (1x1 Mesh):</p> <p>Bits [4:3]: Transaction type</p> <p>2'b00: REQ</p> <p>2'b01: RSP</p> <p>2'b10: SNP</p>	RW	5'b0

por_mxp_p_0-5_syscoreq_ctl

This register repeats 5 times. It parametrized by the index from 0 to 5. Functions as the port #{{index}} snoop and DVM domain control register. Provides a software alternative to hardware SYSCOREQ/SYSCOACK handshake. Works with por_mxp_p#{{index}}_syscoack_status. NOTE: Only valid on RN-F ports.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1C00 + #{16*[0, 1, 2, 3, 4, 5]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mxp_secure_register_groups_override.syscoreq_ctl

The following image shows the higher register bit assignments.



Figure 3-1111 por_mxp_por_mxp_p_0-5_syscoreq_ctl (high)

The following table shows the por_mxp_p_0-5_syscoreq_ctl higher register bit assignments.

Table 3-1131 por_mxp_por_mxp_p_0-5_syscoreq_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

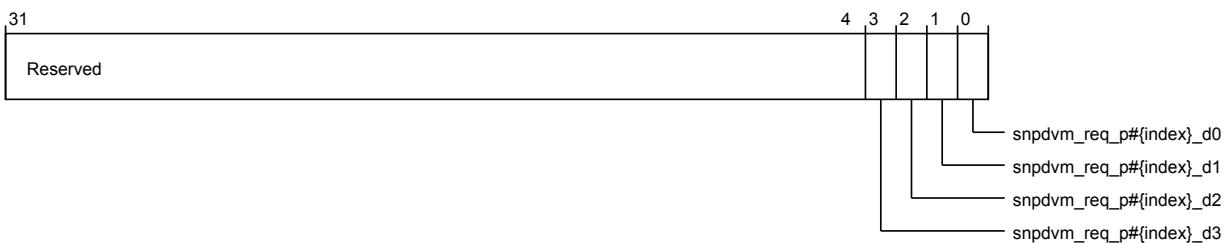


Figure 3-1112 por_mxp_por_mxp_p_0-5_syscoreq_ctl (low)

The following table shows the por_mxp_p_0-5_syscoreq_ctl lower register bit assignments.

Table 3-1132 por_mxp_por_mxp_p_0-5_syscoreq_ctl (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	snpdvm_req_p#{index}_d3	When set, initiates the process of enabling snoop and DVM dispatches (SYSCOREQ) to device 3 on port #{index}	RW	1'b0
2	snpdvm_req_p#{index}_d2	When set, initiates the process of enabling snoop and DVM dispatches (SYSCOREQ) to device 2 on port #{index}	RW	1'b0
1	snpdvm_req_p#{index}_d1	When set, initiates the process of enabling snoop and DVM dispatches (SYSCOREQ) to device 1 on port #{index}	RW	1'b0
0	snpdvm_req_p#{index}_d0	When set, initiates the process of enabling snoop and DVM dispatches (SYSCOREQ) to device 0 on port #{index}	RW	1'b0

por_mxp_p_0-5_syscoack_status

This register repeats 5 times. It parametrized by the index from 0 to 5. Functions as the port #{index} snoop and DVM domain status register. Provides a software alternative to hardware SYSREQ/SYSACK handshake. Works with por_mxp_p#{index}_syscoreq_ctl. NOTE: Only valid on RN-F ports.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h1C08 + #{16*[0, 1, 2, 3, 4, 5]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_mxp_secure_register_groups_override.syscoreq_ctl

The following image shows the higher register bit assignments.

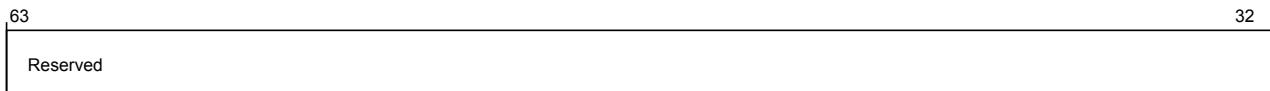


Figure 3-1113 por_mxp_por_mxp_p_0-5_syscoack_status (high)

The following table shows the por_mxp_p_0-5_syscoack_status higher register bit assignments.

Table 3-1133 por_mxp_por_mxp_p_0-5_syscoack_status (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

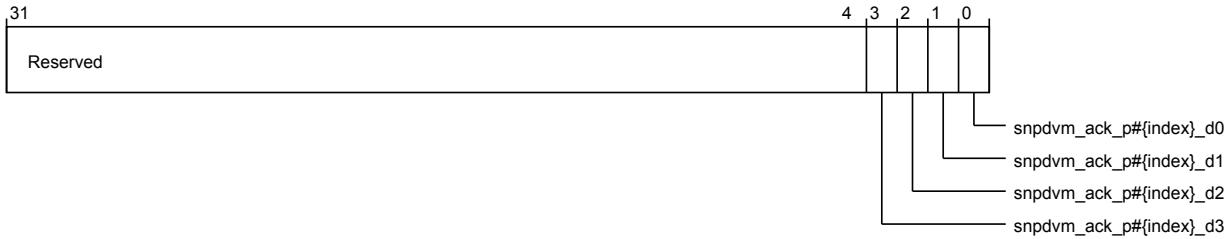


Figure 3-1114 por_mxp_por_mxp_p_0-5_syscoack_status (low)

The following table shows the por_mxp_p_0-5_syscoack_status lower register bit assignments.

Table 3-1134 por_mxp_por_mxp_p_0-5_syscoack_status (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	snpdvm_ack_p#{index}_d3	When set, indicates snoop and DVM dispatches are enabled (SYSCOACK) for device 3 on port #{index}	RO	1'b0
2	snpdvm_ack_p#{index}_d2	When set, indicates snoop and DVM dispatches are enabled (SYSCOACK) for device 2 on port #{index}	RO	1'b0
1	snpdvm_ack_p#{index}_d1	When set, indicates snoop and DVM dispatches are enabled (SYSCOACK) for device 1 on port #{index}	RO	1'b0
0	snpdvm_ack_p#{index}_d0	When set, indicates snoop and DVM dispatches are enabled (SYSCOACK) for device 0 on port #{index}	RO	1'b0

por_dtm_control

Functions as the DTM control register.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2100
Register reset	64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

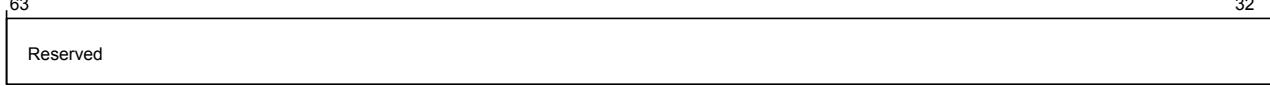


Figure 3-1115 por_mxp_por_dtm_control (high)

The following table shows the por_dtm_control higher register bit assignments.

Table 3-1135 por_mxp_por_dtm_control (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

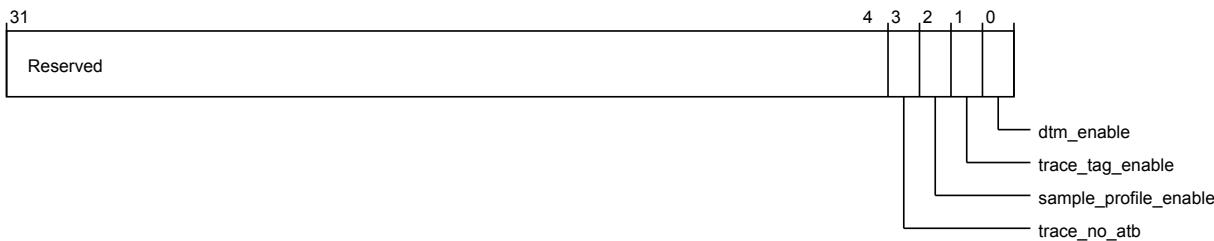


Figure 3-1116 por_mxp_por_dtm_control (low)

The following table shows the por_dtm_control lower register bit assignments.

Table 3-1136 por_mxp_por_dtm_control (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	trace_no_atb	When set, trace packet is not delivered out of ATB, and FIFO entry holds the first trace packet. NOTE: if any MXP has this bit set, ATB protocol will not be functional.	RW	1'b0
2	sample_profile_enable	Enables sample profile function	RW	1'b0
1	trace_tag_enable	Watchpoint trace tag enable 1'b1: Trace tag enabled 1'b0: No trace tag	RW	1'b0
0	dtm_enable	Enables debug watchpoint and PMU function; prior to writing this bit, all other DT configuration registers must be programmed; once this bit is set, other DT configuration registers must not be modified	RW	1'b0

por_dtm_fifo_entry_ready

Controls status of DTM FIFO entries.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h2118
Register reset	64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

Reserved

32

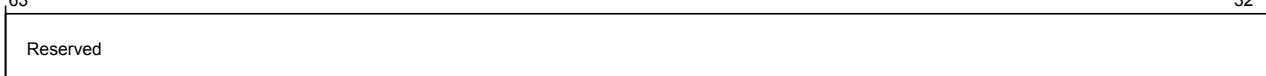


Figure 3-1117 por_mxp_por_dtm_fifo_entry_ready (high)

The following table shows the por_dtm_fifo_entry_ready higher register bit assignments.

Table 3-1137 por_mxp_por_dtm_fifo_entry_ready (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31
Reserved

4 3 0

ready

Figure 3-1118 por_mxp_por_dtm_fifo_entry_ready (low)

The following table shows the por_dtm_fifo_entry_ready lower register bit assignments.

Table 3-1138 por_mxp_por_dtm_fifo_entry_ready (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	ready	Indicates which DTM FIFO entries are ready; write a 1 to clear Bit [3]: Entry 3 ready when set Bit [2]: Entry 2 ready when set Bit [1]: Entry 1 ready when set Bit [0]: Entry 0 ready when set	W1C	4'b0

por_dtm_fifo_entry_0-3_0

This register repeats 3 times. It parametrized by the index from 0 to 3. Contains DTM FIFO entry #{{index}} data.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h2120 + #{{24*[0, 1, 2, 3]}}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	fifo_data0	32
----	------------	----

Figure 3-1119 por_mxp_por_dtm_fifo_entry_0-3_0 (high)

The following table shows the por_dtm_fifo_entry_0-3_0 higher register bit assignments.

Table 3-1139 por_mxp_por_dtm_fifo_entry_0-3_0 (high)

Bits	Field name	Description	Type	Reset
63:32	fifo_data0	Entry data bit vector 63:0	RO	64'b0

The following image shows the lower register bit assignments.

31	fifo_data0	0
----	------------	---

Figure 3-1120 por_mxp_por_dtm_fifo_entry_0-3_0 (low)

The following table shows the por_dtm_fifo_entry_0-3_0 lower register bit assignments.

Table 3-1140 por_mxp_por_dtm_fifo_entry_0-3_0 (low)

Bits	Field name	Description	Type	Reset
31:0	fifo_data0	Entry data bit vector 63:0	RO	64'b0

por_dtm_fifo_entry_0-3_1

This register repeats 3 times. It parametrized by the index from 0 to 3. Contains DTM FIFO entry #{{index}} data.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h2128 + #{{24*[0, 1, 2, 3]}}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63	fifo_data1	32
----	------------	----

Figure 3-1121 por_mxp_por_dtm_fifo_entry_0-3_1 (high)

The following table shows the por_dtm_fifo_entry_0-3_1 higher register bit assignments.

Table 3-1141 por_mxp_por_dtm_fifo_entry_0-3_1 (high)

Bits	Field name	Description	Type	Reset
63:32	fifo_data1	Entry data bit vector 127:64	RO	64'b0

The following image shows the lower register bit assignments.

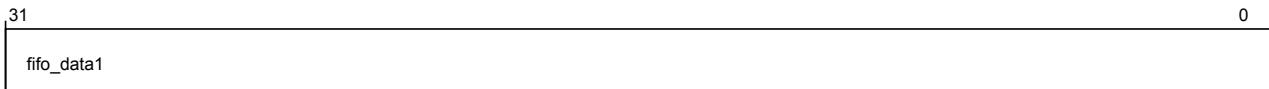


Figure 3-1122 por_mxp_por_dtm_fifo_entry_0-3_1 (low)

The following table shows the por_dtm_fifo_entry_0-3_1 lower register bit assignments.

Table 3-1142 por_mxp_por_dtm_fifo_entry_0-3_1 (low)

Bits	Field name	Description	Type	Reset
31:0	fifo_data1	Entry data bit vector 127:64	RO	64'b0

por_dtm_fifo_entry_0-3_2

This register repeats 3 times. It parametrized by the index from 0 to 3. Contains DTM FIFO entry #{{index}} data.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h2130 + #{24*[0, 1, 2, 3]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

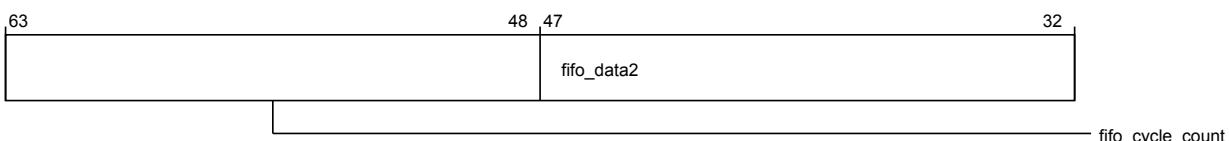


Figure 3-1123 por_mxp_por_dtm_fifo_entry_0-3_2 (high)

The following table shows the por_dtm_fifo_entry_0-3_2 higher register bit assignments.

Table 3-1143 por_mxp_por_dtm_fifo_entry_0-3_2 (high)

Bits	Field name	Description	Type	Reset
63:48	fifo_cycle_count	Entry cycle count bit vector 15:0	RO	16'b0
47:32	fifo_data2	Entry data bit vector 143:128	RO	48'b0

The following image shows the lower register bit assignments.

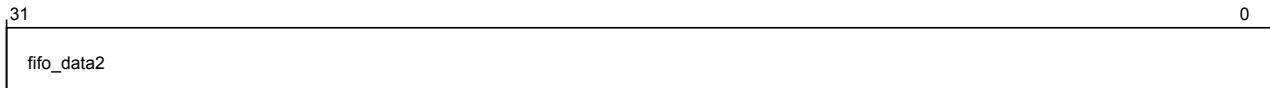


Figure 3-1124 por_mxp_por_dtm_fifo_entry_0-3_2 (low)

The following table shows the por_dtm_fifo_entry_0-3_2 lower register bit assignments.

Table 3-1144 por_mxp_por_dtm_fifo_entry_0-3_2 (low)

Bits	Field name	Description	Type	Reset
31:0	fifo_data2	Entry data bit vector 143:128	RO	48'b0

por_dtm_wp_0-3_config

This register repeats 3 times. It parametrized by the index from 0 to 3. Configures watchpoint # {index}.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h21A0 + # {24*[0, 1, 2, 3]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1125 por_mxp_por_dtm_wp_0-3_config (high)

The following table shows the por_dtm_wp_0-3_config higher register bit assignments.

Table 3-1145 por_mxp_por_dtm_wp_0-3_config (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

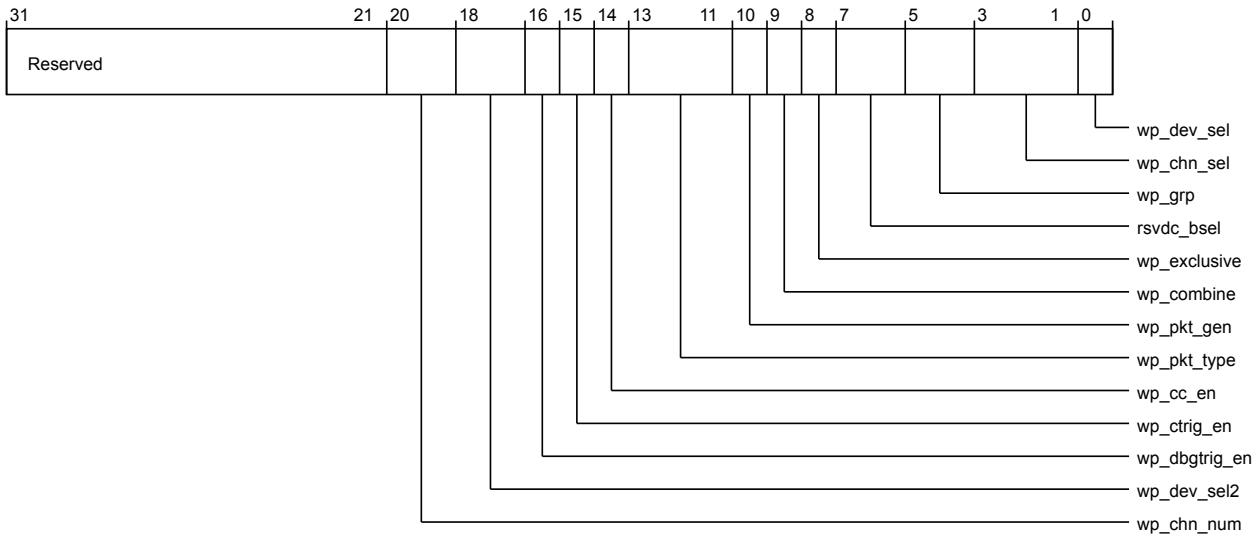


Figure 3-1126 por_mxp_por_dtm_wp_0-3_config (low)

The following table shows the por_dtm_wp_0-3_config lower register bit assignments.

Table 3-1146 por_mxp_por_dtm_wp_0-3_config (low)

Bits	Field name	Description	Type	Reset
31:21	Reserved	Reserved	RO	-
20:19	wp_chn_num	VC number index for replicated channels in specified SMXP	RW	2'b0
18:17	wp_dev_sel2	Upper bits for device port selection in specified SMXP	RW	2'b0
16	wp_dbgtrig_en	Enables watchpoint debug trigger packet generation	RW	1'b0
15	wp_ctrig_en	Enables watchpoint cross trigger packet generation	RW	1'b0
14	wp_cc_en	Enables inclusion of cycle count in watchpoint track packet generation	RW	1'b0
13:11	wp_pkt_type	Trace packet type 3'b000: TXNID (up to X18) 3'b001: TXNID + opcode (up to X9) 3'b010: TXNID + opcode + source ID + target ID (up to X4) 3'b011: Reserved 3'b100: Control flit 3'b101: DAT flit DATA [127:0] 3'b110: DAT flit DATA [255:128] 3'b111: Reserved	RW	3'b000
10	wp_pkt_gen	Enables watchpoint trace packet generation	RW	1'b0
9	wp_combine	Enables combination of watchpoints #{{index}} and #{{index+1}}	RW	1'b0

Table 3-1146 por_mxp_por_dtm_wp_0-3_config (low) (continued)

Bits	Field name	Description	Type	Reset
8	wp_exclusive	Watchpoint mode 1'b0: Regular mode 1'b1: Exclusive mode	RW	1'b0
7:6	rsvdc_bsel	Byte select of RSVDC in trace packet 2'h0: Select RSVDC[7:0] 2'h1: Select RSVDC[15:8] 2'h2: Select RSVDC[23:16] 2'h3: Select RSVDC[31:24]	RW	1'b0
5:4	wp_grp	Watchpoint register format group 2'h0: Select primary group 2'h1: Select secondary group 2'h2: Select tertiary group 2'h3: Reserved	RW	1'b0
3:1	wp_chn_sel	VC selection 3'b000: Select REQ VC 3'b001: Select RSP VC 3'b010: Select SNP VC 3'b011: Select DATA VC NOTE: All other values are reserved.	RW	3'b000
0	wp_dev_sel	Device port selection in specified SMXP 1'b0: Select device port 0 1'b1: Select device port 1	RW	1'b0

por_dtm_wp_0-3_val

This register repeats 3 times. It parametrized by the index from 0 to 3. Configures watchpoint #{{index}} comparison value.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h21A8 + #{{24*[0, 1, 2, 3]}}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

63

val

32

Figure 3-1127 por_mxp_por_dtm_wp_0-3_val (high)

The following table shows the por_dtm_wp_0-3_val higher register bit assignments.

Table 3-1147 por_mxp_por_dtm_wp_0-3_val (high)

Bits	Field name	Description	Type	Reset
63:32	val	Refer to DTM watchpoint section for details	RW	64'b0

The following image shows the lower register bit assignments.

31

val

0

Figure 3-1128 por_mxp_por_dtm_wp_0-3_val (low)

The following table shows the por_dtm_wp_0-3_val lower register bit assignments.

Table 3-1148 por_mxp_por_dtm_wp_0-3_val (low)

Bits	Field name	Description	Type	Reset
31:0	val	Refer to DTM watchpoint section for details	RW	64'b0

por_dtm_wp_0-3_mask

This register repeats 3 times. It parametrized by the index from 0 to 3. Configures watchpoint # {index} comparison mask.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h21B0 + # {24*[0, 1, 2, 3]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

63

mask

32

Figure 3-1129 por_mxp_por_dtm_wp_0-3_mask (high)

The following table shows the por_dtm_wp_0-3_mask higher register bit assignments.

Table 3-1149 por_mxp_por_dtm_wp_0-3_mask (high)

Bits	Field name	Description	Type	Reset
63:32	mask	Refer to DTM watchpoint section for details	RW	64'b0

The following image shows the lower register bit assignments.

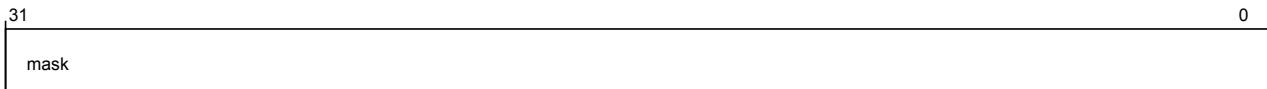


Figure 3-1130 por_mxp_por_dtm_wp_0-3_mask (low)

The following table shows the por_dtm_wp_0-3_mask lower register bit assignments.

Table 3-1150 por_mxp_por_dtm_wp_0-3_mask (low)

Bits	Field name	Description	Type	Reset
31:0	mask	Refer to DTM watchpoint section for details	RW	64'b0

por_dtm_pmsicr

Functions as the sampling interval counter register.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2200

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

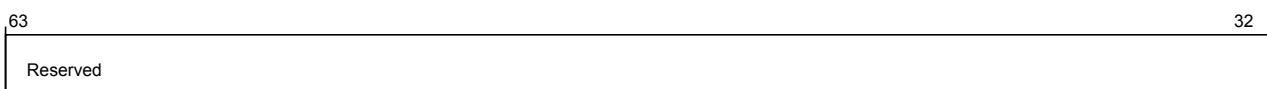


Figure 3-1131 por_mxp_por_dtm_pmsicr (high)

The following table shows the por_dtm_pmsicr higher register bit assignments.

Table 3-1151 por_mxp_por_dtm_pmsicr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

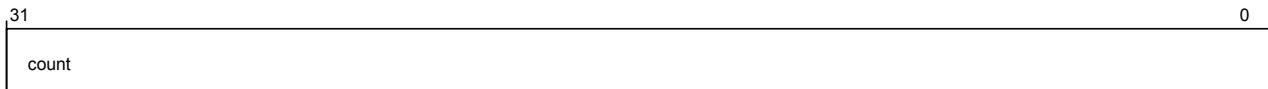


Figure 3-1132 por_mxp_por_dtm_pmsicr (low)

The following table shows the por_dtm_pmsicr lower register bit assignments.

Table 3-1152 por_mxp_por_dtm_pmsicr (low)

Bits	Field name	Description	Type	Reset
31:0	count	Current value of sample counter	RW	32'b0

por_dtm_pmsirr

Functions as the sampling interval reload register.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2208

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

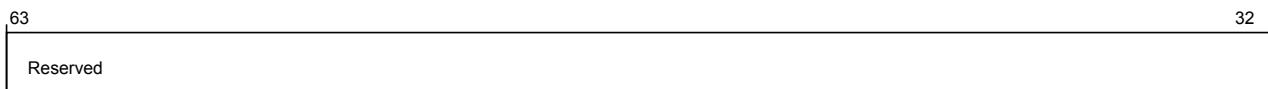


Figure 3-1133 por_mxp_por_dtm_pmsirr (high)

The following table shows the por_dtm_pmsirr higher register bit assignments.

Table 3-1153 por_mxp_por_dtm_pmsirr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

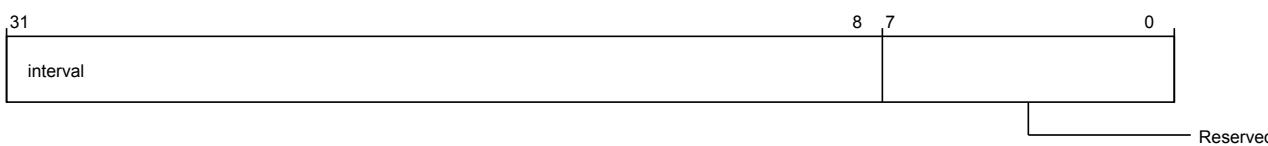


Figure 3-1134 por_mxp_por_dtm_pmsirr (low)

The following table shows the por_dtm_pmsirr lower register bit assignments.

Table 3-1154 por_mxp_por_dtm_pmsirr (low)

Bits	Field name	Description	Type	Reset
31:8	interval	Sampling interval to be reloaded	RW	24'b0
7:0	Reserved	Reserved	RO	-

por_dtm_pmu_config

Configures the DTM PMU.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2210

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

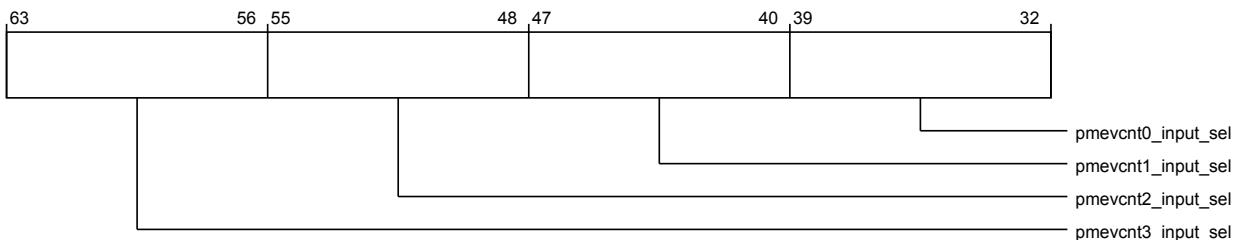


Figure 3-1135 por_mxp_por_dtm_pmu_config (high)

The following table shows the por_dtm_pmu_config higher register bit assignments.

Table 3-1155 por_mxp_por_dtm_pmu_config (high)

Bits	Field name	Description	Type	Reset
63:56	pmevcnt3_input_sel	Source to be counted in PMU counter 3; see pmevcnt0_input_sel for encodings	RW	8'b0
55:48	pmevcnt2_input_sel	Source to be counted in PMU counter 2; see pmevcnt0_input_sel for encodings	RW	8'b0

Table 3-1155 por_mxp_por_dtm_pmu_config (high) (continued)

Bits	Field name	Description	Type	Reset
47:40	pmevcnt1_input_sel	Source to be counted in PMU counter 1; see pmevcnt0_input_sel for encodings	RW	8'b0

Table 3-1155 por_mxp_por_dtm_pmu_config (high) (continued)

Bits	Field name	Description	Type	Reset
39:32	pmevcnt0_input_sel	<p>Source to be counted in PMU counter 0: Port2, Port3, Port4 and Port5 encodings are applicable when (MXP_NUM_DEV_PORT_PARAM > 2 and MXP_MULTIPLE_DTM_EN_PARAM = 0)</p> <p>8'h00: Watchpoint 0 8'h01: Watchpoint 1 8'h02: Watchpoint 2 8'h03: Watchpoint 3 8'h04: XP PMU Event 0 8'h05: XP PMU Event 1 8'h06: XP PMU Event 2 8'h07: XP PMU Event 3 8'h10: Port 0 Device 0 PMU Event 0 8'h11: Port 0 Device 0 PMU Event 1 8'h12: Port 0 Device 0 PMU Event 2 8'h13: Port 0 Device 0 PMU Event 3 8'h14: Port 0 Device 1 PMU Event 0 8'h15: Port 0 Device 1 PMU Event 1 8'h16: Port 0 Device 1 PMU Event 2 8'h17: Port 0 Device 1 PMU Event 3 8'h18: Port 0 Device 2 PMU Event 0 8'h19: Port 0 Device 2 PMU Event 1 8'h1A: Port 0 Device 2 PMU Event 2 8'h1B: Port 0 Device 2 PMU Event 3 8'h1C: Port 0 Device 3 PMU Event 0 8'h1D: Port 0 Device 3 PMU Event 1 8'h1E: Port 0 Device 3 PMU Event 2 8'h1F: Port 0 Device 3 PMU Event 3 8'h20: Port 1 Device 0 PMU Event 0 8'h21: Port 1 Device 0 PMU Event 1 8'h22: Port 1 Device 0 PMU Event 2 8'h23: Port 1 Device 0 PMU Event 3 8'h24: Port 1 Device 1 PMU Event 0 8'h25: Port 1 Device 1 PMU Event 1 8'h26: Port 1 Device 1 PMU Event 2 8'h27: Port 1 Device 1 PMU Event 3 8'h28: Port 1 Device 2 PMU Event 0 8'h29: Port 1 Device 2 PMU Event 1</p> <p>8'h2A: Port 1 Device 2 PMU Event 2</p>	RW	8'b0
102308_0000_01_en		<p>Copyright © 2020 Arm Limited or its affiliates. All rights reserved. Confidential</p> <p>8'h2C: Port 1 Device 3 PMU Event 0</p>		3-1027

The following image shows the lower register bit assignments.

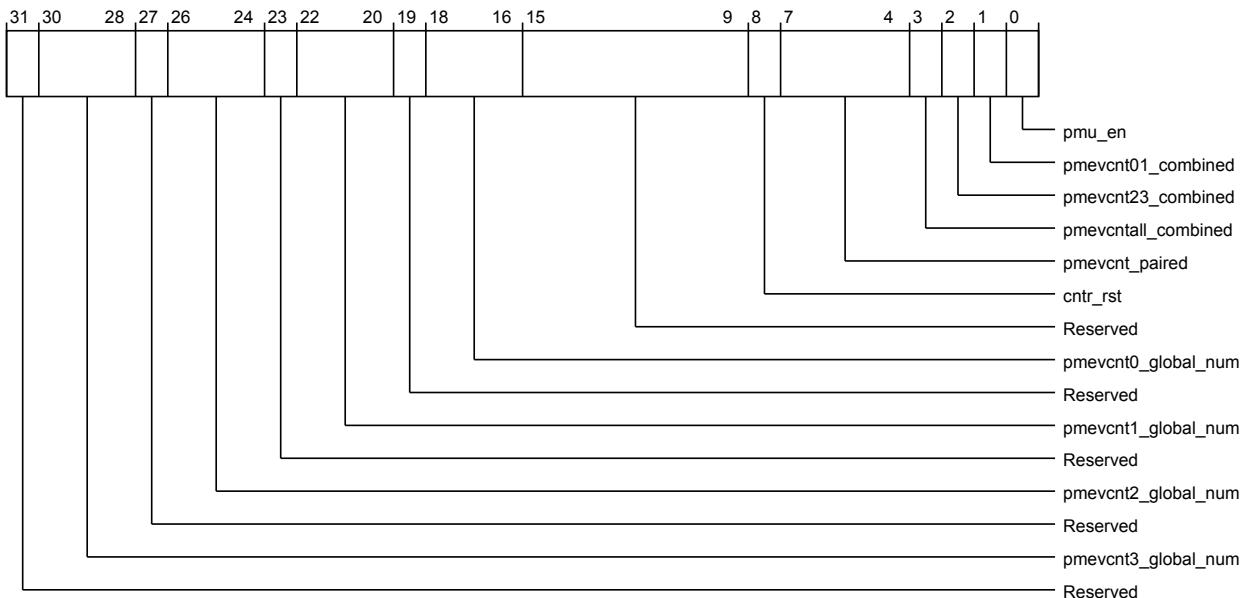


Figure 3-1136 por_mxp_por_dtm_pmu_config (low)

The following table shows the por_dtm_pmu_config lower register bit assignments.

Table 3-1156 por_mxp_por_dtm_pmu_config (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	pmevcnt3_global_num	Global counter to pair with PMU counter 3; see pmevcnt0_global_num for encodings	RW	3'b0
27	Reserved	Reserved	RO	-
26:24	pmevcnt2_global_num	Global counter to pair with PMU counter 2; see pmevcnt0_global_num for encodings	RW	3'b0
23	Reserved	Reserved	RO	-
22:20	pmevcnt1_global_num	Global counter to pair with PMU counter 1; see pmevcnt0_global_num for encodings	RW	3'b0
19	Reserved	Reserved	RO	-
18:16	pmevcnt0_global_num	Global counter to pair with PMU counter 0 3'b000: Global PMU event counter A 3'b001: Global PMU event counter B 3'b010: Global PMU event counter C 3'b011: Global PMU event counter D 3'b100: Global PMU event counter E 3'b101: Global PMU event counter F 3'b110: Global PMU event counter G 3'b111: Global PMU event counter H	RW	3'b0

Table 3-1156 por_mxp_por_dtm_pmu_config (low) (continued)

Bits	Field name	Description	Type	Reset
15:9	Reserved	Reserved	RO	-
8	cntr_RST	Enables clearing of live counters upon assertion of snapshot	RW	1'b0
7:4	pmevcnt_paired	PMU local counter paired with global counter	RW	4'b0
3	pmevcntall_combined	Enables combination of all PMU counters (0, 1, 2, 3) NOTE: When set, pmevcnt01_combined and pmevcnt23_combined have no effect.	RW	1'b0
2	pmevcnt23_combined	Enables combination of PMU counters 2 and 3	RW	1'b0
1	pmevcnt01_combined	Enables combination of PMU counters 0 and 1	RW	1'b0
0	pmu_en	DTM PMU enable NOTE: All other fields in this register are valid only if this bit is set.	RW	1'b0

por_dtm_pmevcnt

Contains all PMU event counters (0, 1, 2, 3).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2220

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

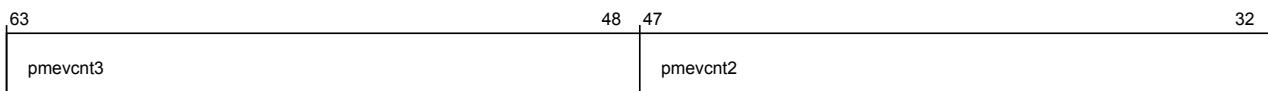


Figure 3-1137 por_mxp_por_dtm_pmevcnt (high)

The following table shows the por_dtm_pmevcnt higher register bit assignments.

Table 3-1157 por_mxp_por_dtm_pmevcnt (high)

Bits	Field name	Description	Type	Reset
63:48	pmevcnt3	PMU event counter 3	RW	16'h0000
47:32	pmevcnt2	PMU event counter 2	RW	16'h0000

The following image shows the lower register bit assignments.

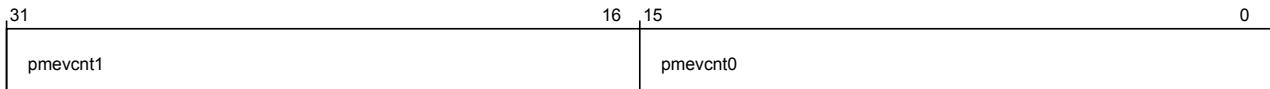


Figure 3-1138 por_mxp_por_dtm_pmevcnt (low)

The following table shows the por_dtm_pmevcnt lower register bit assignments.

Table 3-1158 por_mxp_por_dtm_pmevcnt (low)

Bits	Field name	Description	Type	Reset
31:16	pmevcnt1	PMU event counter 1	RW	16'h0000
15:0	pmevcnt0	PMU event counter 0	RW	16'h0000

por_dtm_pmevcntsr

Functions as the PMU event counter shadow register for all counters (0, 1, 2, 3).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2240

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

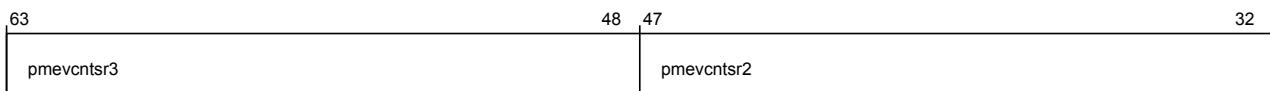


Figure 3-1139 por_mxp_por_dtm_pmevcntsr (high)

The following table shows the por_dtm_pmevcntsr higher register bit assignments.

Table 3-1159 por_mxp_por_dtm_pmevcntsr (high)

Bits	Field name	Description	Type	Reset
63:48	pmevcntsr3	PMU event counter 3 shadow register	RW	16'h0000
47:32	pmevcntsr2	PMU event counter 2 shadow register	RW	16'h0000

The following image shows the lower register bit assignments.

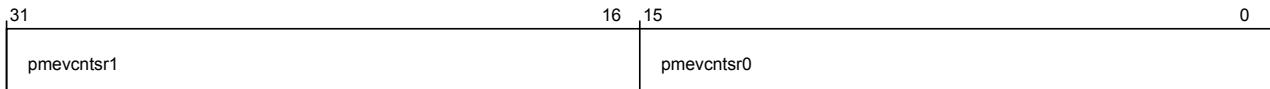


Figure 3-1140 por_mxp_por_dtm_pmevcntsr (low)

The following table shows the por_dtm_pmevcntsr lower register bit assignments.

Table 3-1160 por_mxp_por_dtm_pmevcntsr (low)

Bits	Field name	Description	Type	Reset
31:16	pmevcntsr1	PMU event counter 1 shadow register	RW	16'h0000
15:0	pmevcntsr0	PMU event counter 0 shadow register	RW	16'h0000

por_dtm_control_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Functions as the DTM control register. NOTE: There will be max. of 3 DTM registers based on MXP_MULTIPLE_DTM_EN_PARAM and MXP_NUM_DEV_PORT_PARAM value. Each successive DTM register will be at the next 'h200 + 8 byte address boundary. Each successive DTM register will be named with the suffix corresponding to the DT register number. For example por_dtm_control_dt<0:3>

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2100 + # {512*[1, 2, 3]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

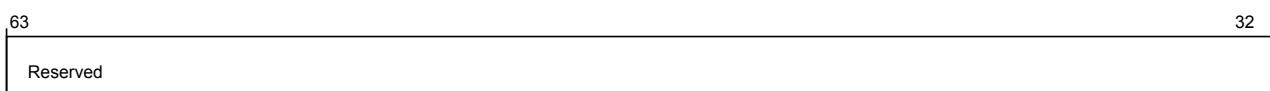


Figure 3-1141 por_mxp_por_dtm_control_dt_1-3 (high)

The following table shows the por_dtm_control_dt_1-3 higher register bit assignments.

Table 3-1161 por_mxp_por_dtm_control_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

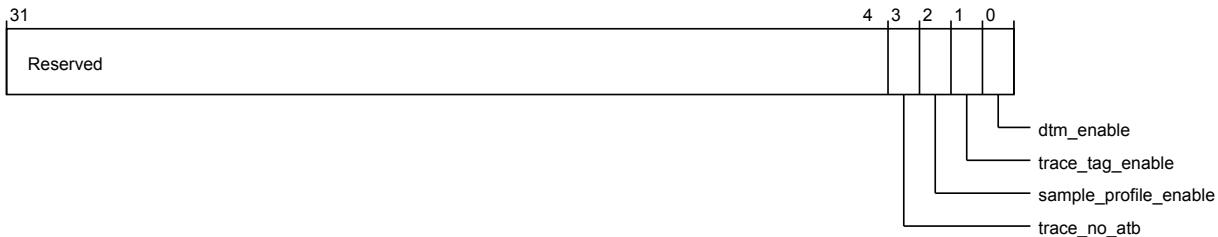


Figure 3-1142 por_mxp_por_dtm_control_dt_1-3 (low)

The following table shows the por_dtm_control_dt_1-3 lower register bit assignments.

Table 3-1162 por_mxp_por_dtm_control_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	trace_no_atb	When set, trace packet is not delivered out of ATB, and FIFO entry holds the first trace packet	RW	1'b0
2	sample_profile_enable	Enables sample profile function	RW	1'b0
1	trace_tag_enable	Watchpoint trace tag enable 1'b1: Trace tag enabled 1'b0: No trace tag	RW	1'b0
0	dtm_enable	Enables debug watchpoint and PMU function; prior to writing this bit, all other DT configuration registers must be programmed; once this bit is set, other DT configuration registers must not be modified	RW	1'b0

por_dtm_fifo_entry_ready_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Controls status of DTM FIFO entries.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h2118 + # {512*[1, 2, 3]}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

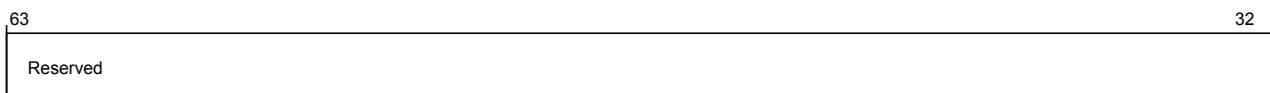


Figure 3-1143 por_mxp_por_dtm_fifo_entry_ready_dt_1-3 (high)

The following table shows the por_dtm_fifo_entry_ready_dt_1-3 higher register bit assignments.

Table 3-1163 por_mxp_por_dtm_fifo_entry_ready_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

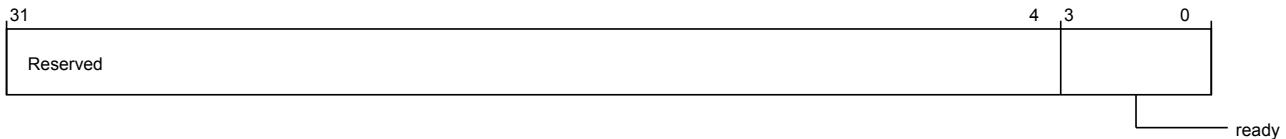


Figure 3-1144 por_mxp_por_dtm_fifo_entry_ready_dt_1-3 (low)

The following table shows the por_dtm_fifo_entry_ready_dt_1-3 lower register bit assignments.

Table 3-1164 por_mxp_por_dtm_fifo_entry_ready_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	ready	Indicates which DTM FIFO entries are ready; write a 1 to clear Bit [3]: Entry 3 ready when set Bit [2]: Entry 2 ready when set Bit [1]: Entry 1 ready when set Bit [0]: Entry 0 ready when set	W1C	4'b0

por_dtm_fifo_entry_0-11

This register repeats 11 times. It parametrized by the index from 0 to 11. Contains DTM FIFO entry $\#\{index\%4\}$ data.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h2130 + $\#\{24*([0, 1, 2, .., 10, 11]\%4)\} + \#\{512*(([0, 1, 2, .., 10, 11]/4)+1)\}$

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

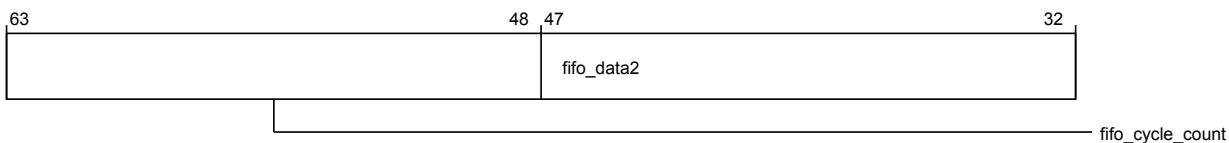


Figure 3-1145 por_mxp_por_dtm_fifo_entry_0-11 (high)

The following table shows the por_dtm_fifo_entry_0-11 higher register bit assignments.

Table 3-1165 por_mxp_por_dtm_fifo_entry_0-11 (high)

Bits	Field name	Description	Type	Reset
63:48	fifo_cycle_count	Entry cycle count bit vector 15:0	RO	16'b0
47:32	fifo_data2	Entry data bit vector 143:128	RO	48'b0

The following image shows the lower register bit assignments.

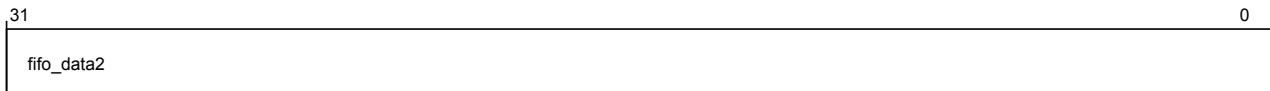


Figure 3-1146 por_mxp_por_dtm_fifo_entry_0-11 (low)

The following table shows the por_dtm_fifo_entry_0-11 lower register bit assignments.

Table 3-1166 por_mxp_por_dtm_fifo_entry_0-11 (low)

Bits	Field name	Description	Type	Reset
31:0	fifo_data2	Entry data bit vector 143:128	RO	48'b0

por_dtm_fifo_entry_0-11

This register repeats 11 times. It parametrized by the index from 0 to 11. Contains DTM FIFO entry $\#\{index\%4\}$ data.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset $16'h2130 + \#\{24*([0, 1, 2, .., 10, 11]\%4)\} + \#\{512*(([0, 1, 2, .., 10, 11]/4)+1)\}$

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

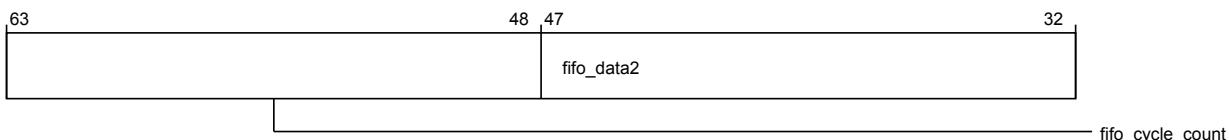


Figure 3-1147 por_mxp_por_dtm_fifo_entry_0-11 (high)

The following table shows the por_dtm_fifo_entry_0-11 higher register bit assignments.

Table 3-1167 por_mxp_por_dtm_fifo_entry_0-11 (high)

Bits	Field name	Description	Type	Reset
63:48	fifo_cycle_count	Entry cycle count bit vector 15:0	RO	16'b0
47:32	fifo_data2	Entry data bit vector 143:128	RO	48'b0

The following image shows the lower register bit assignments.



Figure 3-1148 por_mxp_por_dtm_fifo_entry_0-11 (low)

The following table shows the por_dtm_fifo_entry_0-11 lower register bit assignments.

Table 3-1168 por_mxp_por_dtm_fifo_entry_0-11 (low)

Bits	Field name	Description	Type	Reset
31:0	fifo_data2	Entry data bit vector 143:128	RO	48'b0

por_dtm_fifo_entry_0-11

This register repeats 11 times. It parametrized by the index from 0 to 11. Contains DTM FIFO entry $\#\{index\%4\}$ data.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset $16'h2130 + \#\{24*([0, 1, 2, .., 10, 11]\%4)\} + \#\{512*(([0, 1, 2, .., 10, 11]/4)+1)\}$

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

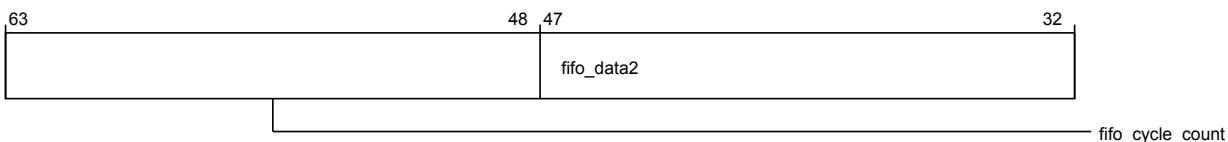


Figure 3-1149 por_mxp_por_dtm_fifo_entry_0-11 (high)

The following table shows the por_dtm_fifo_entry_0-11 higher register bit assignments.

Table 3-1169 por_mxp_por_dtm_fifo_entry_0-11 (high)

Bits	Field name	Description	Type	Reset
63:48	fifo_cycle_count	Entry cycle count bit vector 15:0	RO	16'b0
47:32	fifo_data2	Entry data bit vector 143:128	RO	48'b0

The following image shows the lower register bit assignments.



Figure 3-1150 por_mxp_por_dtm_fifo_entry_0-11 (low)

The following table shows the por_dtm_fifo_entry_0-11 lower register bit assignments.

Table 3-1170 por_mxp_por_dtm_fifo_entry_0-11 (low)

Bits	Field name	Description	Type	Reset
31:0	fifo_data2	Entry data bit vector 143:128	RO	48'b0

por_dtm_wp_0-11

This register repeats 11 times. It parametrized by the index from 0 to 11. Configures watchpoint #{{index} %4} comparison mask.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h21B0 + #{{24*([0, 1, 2, .., 10, 11] % 4)} + #{{512*(([0, 1, 2, .., 10, 11]/4)+1)}}}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

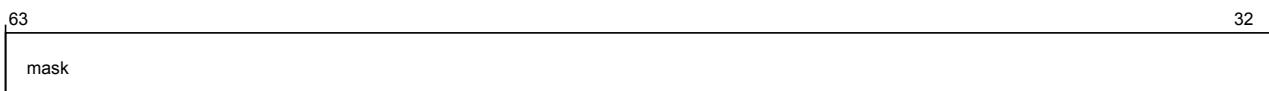


Figure 3-1151 por_mxp_por_dtm_wp_0-11 (high)

The following table shows the por_dtm_wp_0-11 higher register bit assignments.

Table 3-1171 por_mxp_por_dtm_wp_0-11 (high)

Bits	Field name	Description	Type	Reset
63:32	mask	Refer to DTM watchpoint section for details	RW	64'b0

The following image shows the lower register bit assignments.

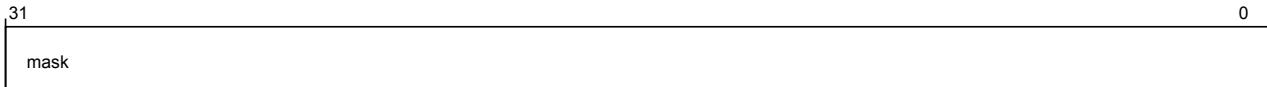


Figure 3-1152 por_mxp_por_dtm_wp_0-11 (low)

The following table shows the por_dtm_wp_0-11 lower register bit assignments.

Table 3-1172 por_mxp_por_dtm_wp_0-11 (low)

Bits	Field name	Description	Type	Reset
31:0	mask	Refer to DTM watchpoint section for details	RW	64'b0

por_dtm_wp_0-11

This register repeats 11 times. It parametrized by the index from 0 to 11. Configures watchpoint #{{index %4}} comparison mask.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h21B0 + #{24*([0, 1, 2, .., 10, 11]%4)} + #{512*(([0, 1, 2, .., 10, 11]/4)+1)}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

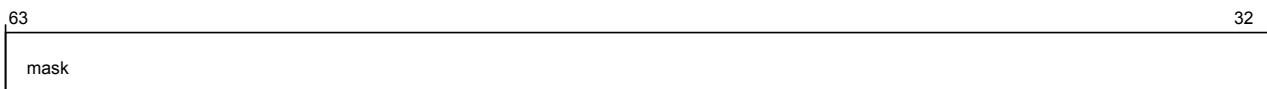


Figure 3-1153 por_mxp_por_dtm_wp_0-11 (high)

The following table shows the por_dtm_wp_0-11 higher register bit assignments.

Table 3-1173 por_mxp_por_dtm_wp_0-11 (high)

Bits	Field name	Description	Type	Reset
63:32	mask	Refer to DTM watchpoint section for details	RW	64'b0

The following image shows the lower register bit assignments.



Figure 3-1154 por_mxp_por_dtm_wp_0-11 (low)

The following table shows the por_dtm_wp_0-11 lower register bit assignments.

Table 3-1174 por_mxp_por_dtm_wp_0-11 (low)

Bits	Field name	Description	Type	Reset
31:0	mask	Refer to DTM watchpoint section for details	RW	64'b0

por_dtm_wp_0-11

This register repeats 11 times. It parametrized by the index from 0 to 11. Configures watchpoint #{{index %4} comparison mask.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h21B0 + #{24*([0, 1, 2, .., 10, 11]%4)} + #{512*(([0, 1, 2, .., 10, 11]/4)+1)}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

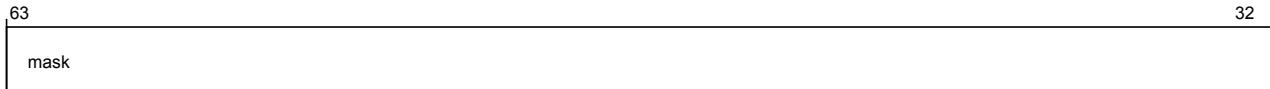


Figure 3-1155 por_mxp_por_dtm_wp_0-11 (high)

The following table shows the por_dtm_wp_0-11 higher register bit assignments.

Table 3-1175 por_mxp_por_dtm_wp_0-11 (high)

Bits	Field name	Description	Type	Reset
63:32	mask	Refer to DTM watchpoint section for details	RW	64'b0

The following image shows the lower register bit assignments.

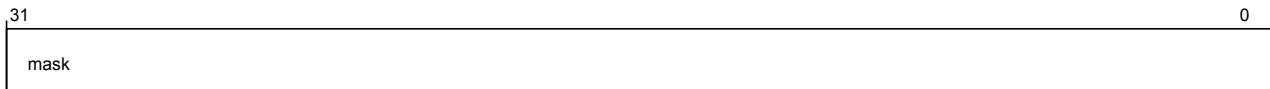


Figure 3-1156 por_mxp_por_dtm_wp_0-11 (low)

The following table shows the por_dtm_wp_0-11 lower register bit assignments.

Table 3-1176 por_mxp_por_dtm_wp_0-11 (low)

Bits	Field name	Description	Type	Reset
31:0	mask	Refer to DTM watchpoint section for details	RW	64'b0

por_dtm_pmsicr_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Functions as the sampling interval counter register.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2200 + #{512*[1, 2, 3]}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1157 por_mxp_por_dtm_pmsicr_dt_1-3 (high)

The following table shows the por_dtm_pmsicr_dt_1-3 higher register bit assignments.

Table 3-1177 por_mxp_por_dtm_pmsicr_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

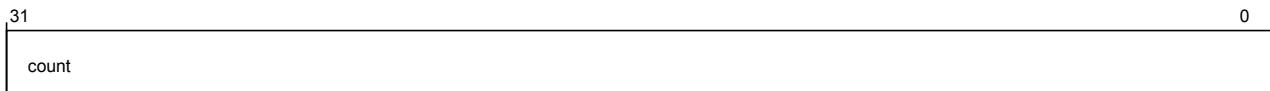


Figure 3-1158 por_mxp_por_dtm_pmsicr_dt_1-3 (low)

The following table shows the por_dtm_pmsicr_dt_1-3 lower register bit assignments.

Table 3-1178 por_mxp_por_dtm_pmsicr_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31:0	count	Current value of sample counter	RW	32'b0

por_dtm_pmsicr_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Functions as the sampling interval reload register.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2208 + #{512*[1, 2, 3]}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

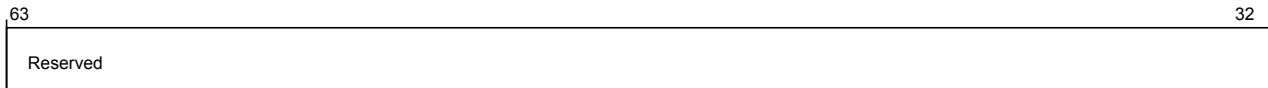


Figure 3-1159 por_mxp_por_dtm_pmsirr_dt_1-3 (high)

The following table shows the por_dtm_pmsirr_dt_1-3 higher register bit assignments.

Table 3-1179 por_mxp_por_dtm_pmsirr_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

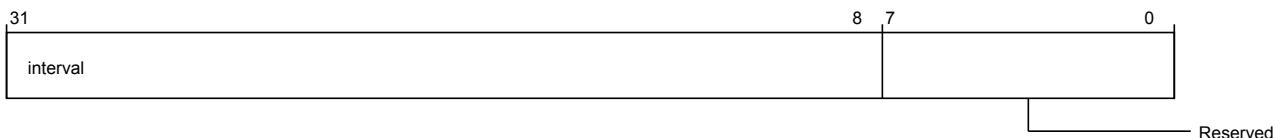


Figure 3-1160 por_mxp_por_dtm_pmsirr_dt_1-3 (low)

The following table shows the por_dtm_pmsirr_dt_1-3 lower register bit assignments.

Table 3-1180 por_mxp_por_dtm_pmsirr_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31:8	interval	Sampling interval to be reloaded	RW	24'b0
7:0	Reserved	Reserved	RO	-

por_dtm_pmu_config_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Configures the DTM PMU.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2210 + # {512*[1, 2, 3]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

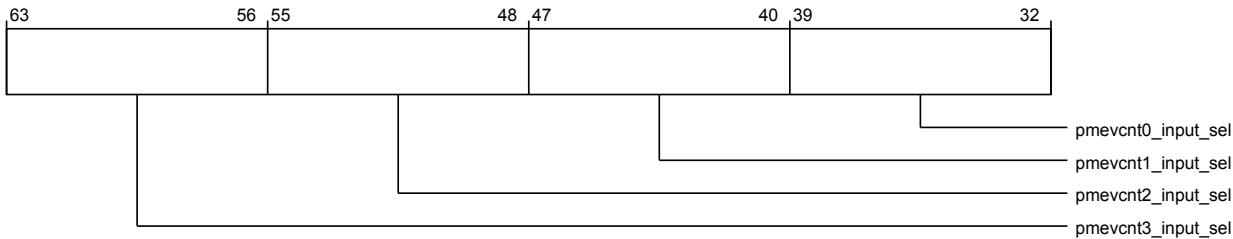


Figure 3-1161 por_mxp_por_dtm_pmu_config_dt_1-3 (high)

The following table shows the por_dtm_pmu_config_dt_1-3 higher register bit assignments.

Table 3-1181 por_mxp_por_dtm_pmu_config_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:56	pmevcnt3_input_sel	Source to be counted in PMU counter 3; see pmevcnt0_input_sel for encodings	RW	8'b0
55:48	pmevcnt2_input_sel	Source to be counted in PMU counter 2; see pmevcnt0_input_sel for encodings	RW	8'b0

Table 3-1181 por_mxp_por_dtm_pmu_config_dt_1-3 (high) (continued)

Bits	Field name	Description	Type	Reset
47:40	pmevcnt1_input_sel	Source to be counted in PMU counter 1; see pmevcnt0_input_sel for encodings	RW	8'b0

Table 3-1181 por_mxp_por_dtm_pmu_config_dt_1-3 (high) (continued)

Bits	Field name	Description	Type	Reset
39:32	pmevcnt0_input_sel	<p>Source to be counted in PMU counter 0: Supports 2 Ports (DT1: P2 and P3, DT2: P4 and P5) when (MXP_NUM_DEV_PORT_PARAM > 2 and MXP_MULTIPLE_DTM_EN_PARAM = 1)</p> <p>8'h00: Watchpoint 0 8'h01: Watchpoint 1 8'h02: Watchpoint 2 8'h03: Watchpoint 3 8'h04: XP PMU Event 0 8'h05: XP PMU Event 1 8'h06: XP PMU Event 2 8'h07: XP PMU Event 3 8'h10: Port 0 Device 0 PMU Event 0 8'h11: Port 0 Device 0 PMU Event 1 8'h12: Port 0 Device 0 PMU Event 2 8'h13: Port 0 Device 0 PMU Event 3 8'h14: Port 0 Device 1 PMU Event 0 8'h15: Port 0 Device 1 PMU Event 1 8'h16: Port 0 Device 1 PMU Event 2 8'h17: Port 0 Device 1 PMU Event 3 8'h18: Port 0 Device 2 PMU Event 0 8'h19: Port 0 Device 2 PMU Event 1 8'h1A: Port 0 Device 2 PMU Event 2 8'h1B: Port 0 Device 2 PMU Event 3 8'h1C: Port 0 Device 3 PMU Event 0 8'h1D: Port 0 Device 3 PMU Event 1 8'h1E: Port 0 Device 3 PMU Event 2 8'h1F: Port 0 Device 3 PMU Event 3 8'h20: Port 1 Device 0 PMU Event 0 8'h21: Port 1 Device 0 PMU Event 1 8'h22: Port 1 Device 0 PMU Event 2 8'h23: Port 1 Device 0 PMU Event 3 8'h24: Port 1 Device 1 PMU Event 0 8'h25: Port 1 Device 1 PMU Event 1 8'h26: Port 1 Device 1 PMU Event 2 8'h27: Port 1 Device 1 PMU Event 3 8'h28: Port 1 Device 2 PMU Event 0 8'h29: Port 1 Device 2 PMU Event 1</p> <p>8'h2A: Port 1 Device 2 PMU Event 2 8'h2B: Port 1 Device 2 PMU Event 3 8'h2C: Port 1 Device 3 PMU Event 0 8'h2D: Port 1 Device 3 PMU Event 1</p>	RW	8'b0
102308_0000_01_en				

The following image shows the lower register bit assignments.

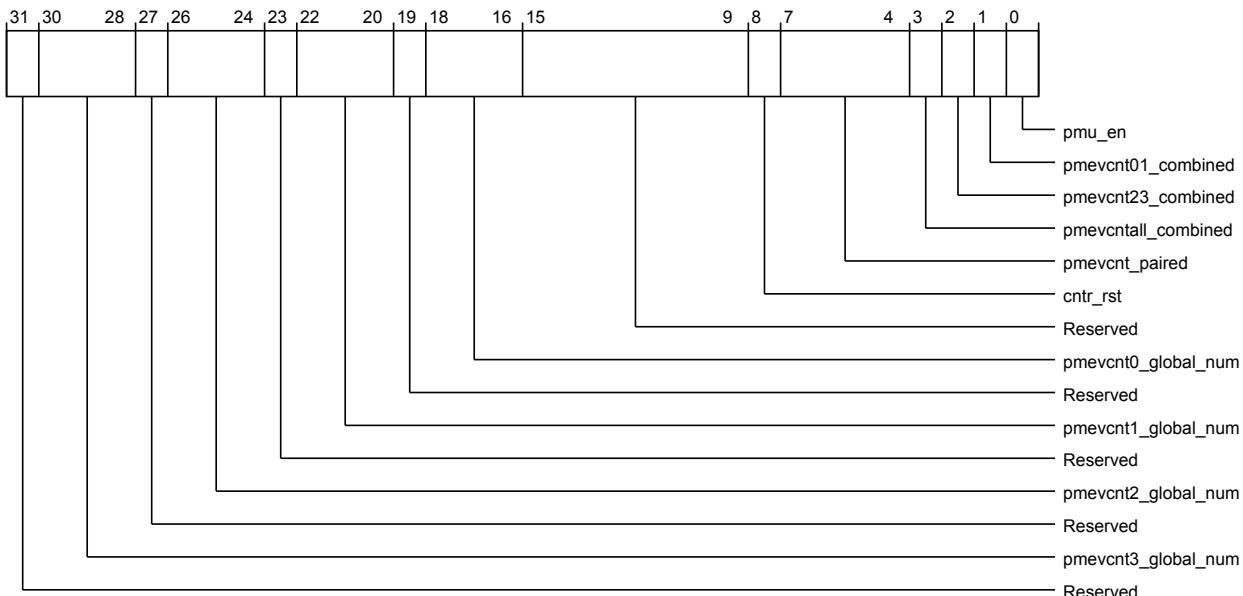


Figure 3-1162 por_mxp_por_dtm_pmu_config_dt_1-3 (low)

The following table shows the por_dtm_pmu_config_dt_1-3 lower register bit assignments.

Table 3-1182 por_mxp_por_dtm_pmu_config_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31	Reserved	Reserved	RO	-
30:28	pmevcnt3_global_num	Global counter to pair with PMU counter 3; see pmevcnt0_global_num for encodings	RW	3'b0
27	Reserved	Reserved	RO	-
26:24	pmevcnt2_global_num	Global counter to pair with PMU counter 2; see pmevcnt0_global_num for encodings	RW	3'b0
23	Reserved	Reserved	RO	-
22:20	pmevcnt1_global_num	Global counter to pair with PMU counter 1; see pmevcnt0_global_num for encodings	RW	3'b0
19	Reserved	Reserved	RO	-
18:16	pmevcnt0_global_num	Global counter to pair with PMU counter 0 3'b000: Global PMU event counter A 3'b001: Global PMU event counter B 3'b010: Global PMU event counter C 3'b011: Global PMU event counter D 3'b100: Global PMU event counter E 3'b101: Global PMU event counter F 3'b110: Global PMU event counter G 3'b111: Global PMU event counter H	RW	3'b0

Table 3-1182 por_mxp_por_dtm_pmu_config_dt_1-3 (low) (continued)

Bits	Field name	Description	Type	Reset
15:9	Reserved	Reserved	RO	-
8	cntr_RST	Enables clearing of live counters upon assertion of snapshot	RW	1'b0
7:4	pmevcnt_paired	PMU local counter paired with global counter	RW	4'b0
3	pmevcntall_combined	Enables combination of all PMU counters (0, 1, 2, 3) NOTE: When set, pmevcnt01_combined and pmevcnt23_combined have no effect.	RW	1'b0
2	pmevcnt23_combined	Enables combination of PMU counters 2 and 3	RW	1'b0
1	pmevcnt01_combined	Enables combination of PMU counters 0 and 1	RW	1'b0
0	pmu_en	DTM PMU enable NOTE: All other fields in this register are valid only if this bit is set.	RW	1'b0

por_dtm_pmevcnt_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Contains all PMU event counters (0, 1, 2, 3).

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2220 + # {512*[1, 2, 3]}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

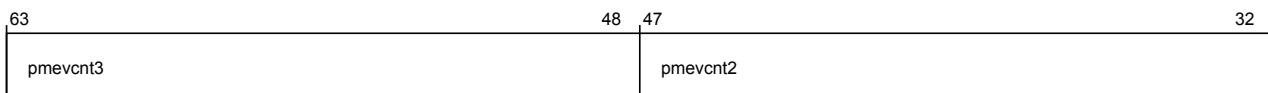


Figure 3-1163 por_mxp_por_dtm_pmevcnt_dt_1-3 (high)

The following table shows the por_dtm_pmevcnt_dt_1-3 higher register bit assignments.

Table 3-1183 por_mxp_por_dtm_pmevcnt_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:48	pmevcnt3	PMU event counter 3	RW	16'h0000
47:32	pmevcnt2	PMU event counter 2	RW	16'h0000

The following image shows the lower register bit assignments.

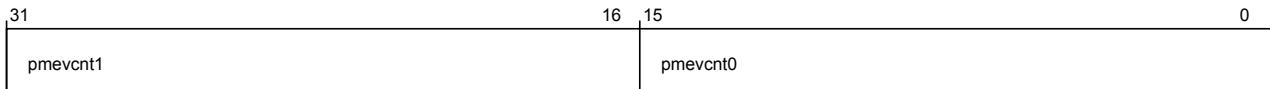


Figure 3-1164 por_mxp_por_dtm_pmevcnt_dt_1-3 (low)

The following table shows the por_dtm_pmevcnt_dt_1-3 lower register bit assignments.

Table 3-1184 por_mxp_por_dtm_pmevcnt_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31:16	pmevcnt1	PMU event counter 1	RW	16'h0000
15:0	pmevcnt0	PMU event counter 0	RW	16'h0000

por_dtm_pmevcntsr_dt_1-3

This register repeats 2 times. It parametrized by the index from 1 to 3. Functions as the PMU event counter shadow register for all counters (0, 1, 2, 3).

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h2240 + # {512*[1, 2, 3]}
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.

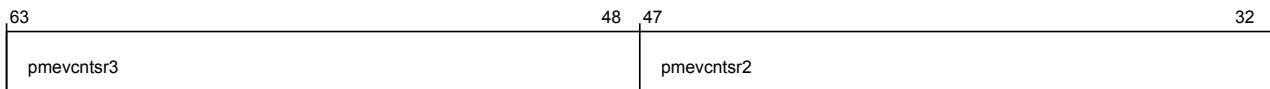


Figure 3-1165 por_mxp_por_dtm_pmevcntsr_dt_1-3 (high)

The following table shows the por_dtm_pmevcntsr_dt_1-3 higher register bit assignments.

Table 3-1185 por_mxp_por_dtm_pmevcntsr_dt_1-3 (high)

Bits	Field name	Description	Type	Reset
63:48	pmevcntsr3	PMU event counter 3 shadow register	RW	16'h0000
47:32	pmevcntsr2	PMU event counter 2 shadow register	RW	16'h0000

The following image shows the lower register bit assignments.

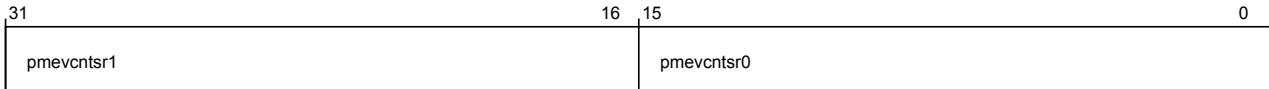


Figure 3-1166 por_mxp_por_dtm_pmevcnts_dt_1-3 (low)

The following table shows the por_dtm_pmevcntsr_dt_1-3 lower register bit assignments.

Table 3-1186 por_mxp_por_dtm_pmevcnts_r_dt_1-3 (low)

Bits	Field name	Description	Type	Reset
31:16	pmevcntsr1	PMU event counter 1 shadow register	RW	16'h0000
15:0	pmeventsrs0	PMU event counter 0 shadow register	RW	16'h0000

por_mxp_multi_mesh_chn_sel_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Functions as the CHI VC channel select per Target register in Multi-Mesh Channel structure.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00 + #{8*[0, 1, 2, .., 14, 15]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mxp_secure_register_groups_override.multi_mesh_ctl

The following image shows the higher register bit assignments.

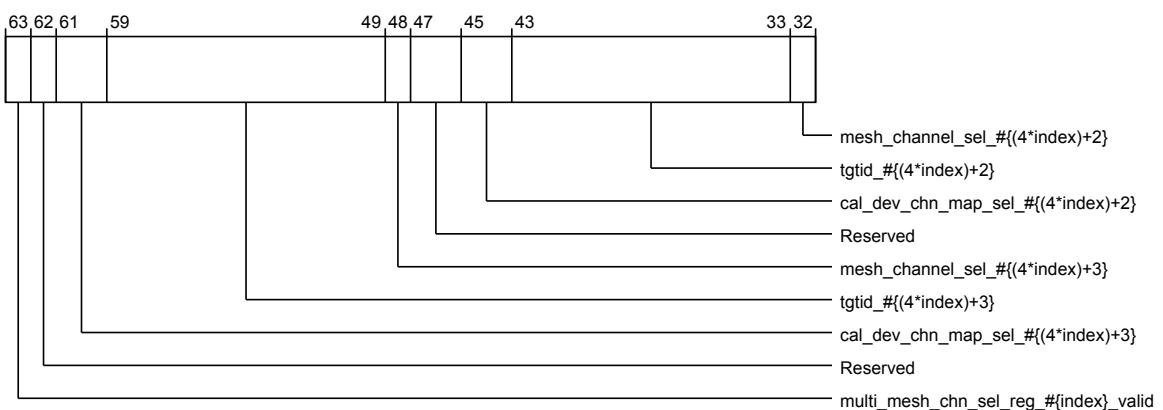


Figure 3-1167 por mxp por mxp multi mesh chn sel 0-15 (high)

The following table shows the port map for multi mesh channel selection 0-15 higher register bit assignments.

Table 3-1187 por_mxp_por_mxp_multi_mesh_chn_sel_0-15 (high)

Bits	Field name	Description	Type	Reset
63	multi_mesh_chn_sel_reg_{index}_valid	Indicates that multi mesh CHI VC channel configured for the targets specified in this register.	RW	1'b0
62	Reserved	Reserved	RO	-
61:60	cal_dev_chn_map_sel_{(4*index)+3}	<p>Channel Map select for target devices behind CAL (associated with the corresponding tgtid field):</p> <p>2'b00: CAL2: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below (applicable for Mesh with > 2 device ports per XP),</p> <ul style="list-style-type: none"> - CAL2: DEV0,DEV1 are mapped to same channel, 2'b01: CAL4: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below, - CAL4: DEV0,DEV1,DEV2,DEV3 are mapped to same channel, 2'b10: Reserved, 2'b11: Each target device behind CAL can be mapped to different channel in the Multi-Channel Mesh structure as specified below, - CAL2: DEV0,DEV1 can be mapped to different channel, - CAL4: DEV0,DEV1,DEV2,DEV3 can be mapped to different channel 	RW	2'b0
59:49	tgtid_{(4*index)+3}	11-bit Target ID associated with the corresponding channel_sel field. This field is used in the LUP to determine which CHI VC channel the FLIT has to be routed to for this target.	RW	11'b0
48	mesh_channel_sel_{(4*index)+3}	CHI VC channel select: 1 - CHI VC channel 1 is selected, 0 - CHI VC channel 0 is selected	RW	1'b0
47:46	Reserved	Reserved	RO	-
45:44	cal_dev_chn_map_sel_{(4*index)+2}	<p>Channel Map select for target devices behind CAL (associated with the corresponding tgtid field):</p> <p>2'b00: CAL2: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below (applicable for Mesh with > 2 device ports per XP),</p> <ul style="list-style-type: none"> - CAL2: DEV0,DEV1 are mapped to same channel, 2'b01: CAL4: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below, - CAL4: DEV0,DEV1,DEV2,DEV3 are mapped to same channel, 2'b10: Reserved, 2'b11: Each target device behind CAL can be mapped to different channel in the Multi-Channel Mesh structure as specified below, - CAL2: DEV0,DEV1 can be mapped to different channel, - CAL4: DEV0,DEV1,DEV2,DEV3 can be mapped to different channel 	RW	1'b0

Table 3-1187 por_mxp_por_mxp_multi_mesh_chn_sel_0-15 (high) (continued)

Bits	Field name	Description	Type	Reset
43:33	tgtid_{(4*index)+2}	11-bit Target ID associated with the corresponding channel_sel field. This field is used in the LUP to determine which CHI VC channel the FLIT has to be routed to for this target.	RW	11'b0
32	mesh_channel_sel_{(4*index)+2}	CHI VC channel select: 1 - CHI VC channel 1 is selected, 0 - CHI VC channel 0 is selected	RW	1'b0

The following image shows the lower register bit assignments.

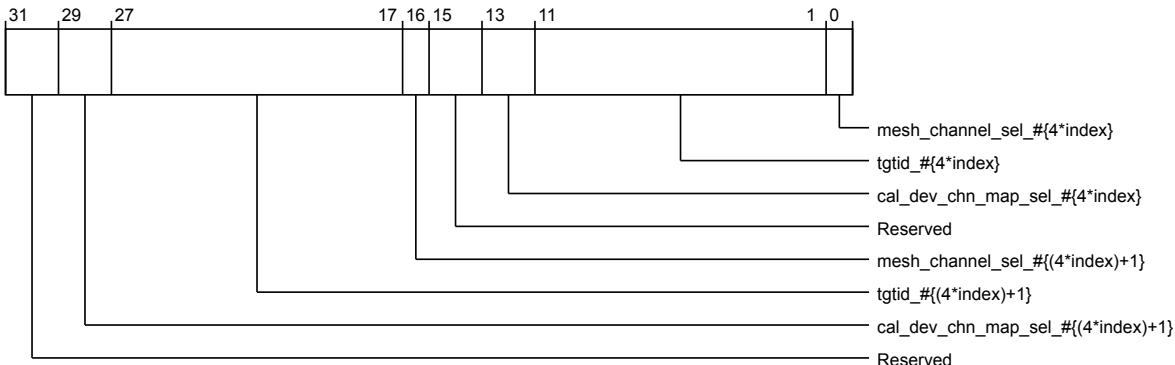


Figure 3-1168 por_mxp_por_mxp_multi_mesh_chn_sel_0-15 (low)

The following table shows the por_mxp_por_mxp_multi_mesh_chn_sel_0-15 lower register bit assignments.

Table 3-1188 por_mxp_por_mxp_multi_mesh_chn_sel_0-15 (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:28	cal_dev_chn_map_sel_{(4*index)+1}	Channel Map select for target devices behind CAL (associated with the corresponding tgtid field): 2'b00: CAL2: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below (applicable for Mesh with > 2 device ports per XP), - CAL2: DEV0,DEV1 are mapped to same channel, 2'b01: CAL4: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below, - CAL4: DEV0,DEV1,DEV2,DEV3 are mapped to same channel, 2'b10: Reserved, 2'b11: Each target device behind CAL can be mapped to different channel in the Multi-Channel Mesh structure as specified below, - CAL2: DEV0,DEV1 can be mapped to different channel, - CAL4: DEV0,DEV1,DEV2,DEV3 can be mapped to different channel	RW	2'b0

Table 3-1188 por_mxp_por_mxp_multi_mesh_chn_sel_0-15 (low) (continued)

Bits	Field name	Description	Type	Reset
27:17	tgtid_{(4*index)+1}	11-bit Target ID associated with the corresponding channel_sel field. This field is used in the LUP to determine which CHI VC channel the FLIT has to be routed to for this target.	RW	11'b0
16	mesh_channel_sel_{(4*index)+1}	CHI VC channel select: 1 - CHI VC channel 1 is selected, 0 - CHI VC channel 0 is selected	RW	1'b0
15:14	Reserved	Reserved	RO	-
13:12	cal_dev_chn_map_sel_{4*index}	Channel Map select for target devices behind CAL (associated with the corresponding tgtid field): 2'b00: CAL2: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below (applicable for Mesh with > 2 device ports per XP), - CAL2: DEV0,DEV1 are mapped to same channel, 2'b01: CAL4: All devices behind CAL are mapped to same channel in the Multi-Channel Mesh structure as specified below, - CAL4: DEV0,DEV1,DEV2,DEV3 are mapped to same channel, 2'b10: Reserved, 2'b11: Each target device behind CAL can be mapped to different channel in the Multi-Channel Mesh structure as specified below, - CAL2: DEV0,DEV1 can be mapped to different channel, - CAL4: DEV0,DEV1,DEV2,DEV3 can be mapped to different channel	RW	2'b0
11:1	tgtid_{4*index}	11-bit Target ID associated with the corresponding channel_sel field. This field is used in the LUP to determine which CHI VC channel the FLIT has to be routed to for this target.	RW	11'b0
0	mesh_channel_sel_{4*index}	CHI VC channel select: 1 - CHI VC channel 1 is selected, 0 - CHI VC channel 0 is selected	RW	1'b0

por_mxp_multi_mesh_chn_ctrl

Functions as the control register for Target based channel selection in Multi-Mesh Channel structure.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC80
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mxp_secure_register_groups_override.multi_mesh_ctl

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-1169 por_mxp_por_mxp_multi_mesh_chn_ctrl (high)

The following table shows the por_mxp_multi_mesh_chn_ctrl higher register bit assignments.

Table 3-1189 por_mxp_por_mxp_multi_mesh_chn_ctrl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

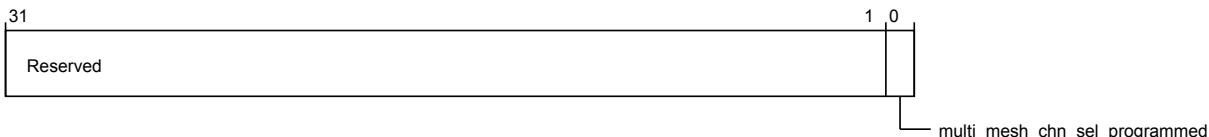


Figure 3-1170 por_mxp_por_mxp_multi_mesh_chn_ctrl (low)

The following table shows the por_mxp_multi_mesh_chn_ctrl lower register bit assignments.

Table 3-1190 por_mxp_por_mxp_multi_mesh_chn_ctrl (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	multi_mesh_chn_sel_programmed	Indicates that multi CHI VC channel configured for all the targets specified in the channel select registers.	RW	1'b0

por_mxp_xy_override_sel_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Functions as SRC-TGT pair whose X-Y route path can be overriden (for Non-XY Route feature per Souce-Target pair) per XP.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC90 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_mxp_secure_register_groups_override.xy_override_ctl

The following image shows the higher register bit assignments.

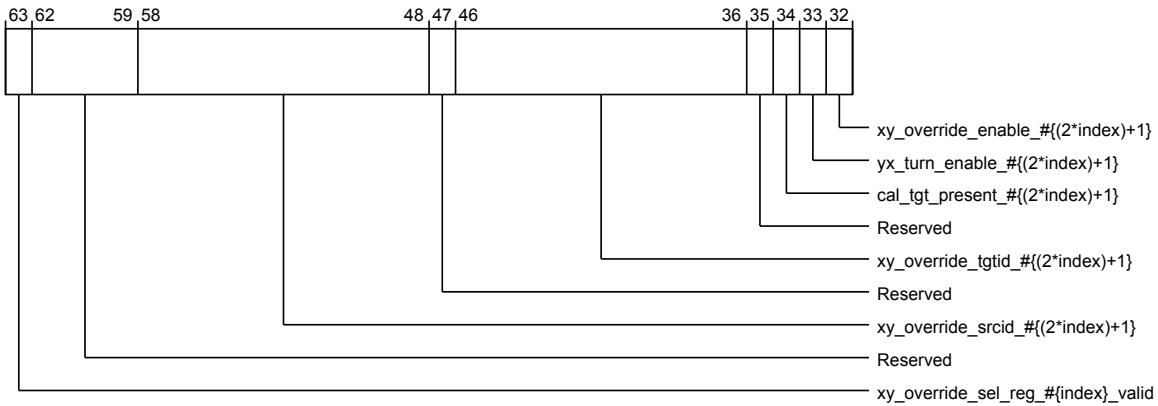


Figure 3-1171 por_mxp_por_mxp_xy_override_sel_0-7 (high)

The following table shows the por_mxp_xy_override_sel_0-7 higher register bit assignments.

Table 3-1191 por_mxp_por_mxp_xy_override_sel_0-7 (high)

Bits	Field name	Description	Type	Reset
63	xy_override_sel_reg_{index}_valid	Indicates that Source-Target pairs whose X-Y route path can be overriden are configured in this register.	RW	1'b0
62:59	Reserved	Reserved	RO	-
58:48	xy_override_srcid_{(2*index)+1}	11-bit Source ID associated with the XY Override. This field is used in the LUP to determine if XY route for the associated source-target pair needs to be overriden.	RW	11'b0
47	Reserved	Reserved	RO	-
46:36	xy_override_tgtid_{(2*index)+1}	11-bit Target ID associated with the XY Override. This field is used in the LUP to determine if XY route for the associated source-target pair needs to be overriden.	RW	11'b0
35	Reserved	Reserved	RO	-
34	cal_tgt_present_{(2*index)+1}	CAL TGT Presence Indication for XY Route Override of all devices behind CAL: 1 - CAL4 TGT Present for XY Route Override of all devices behind CAL, 0 - CAL2 TGT or no CAL TGT Present for XY Route Override of all devices behind CAL.	RW	1'b0
33	yx_turn_enable_{(2*index)+1}	Y-X Turn Enable: 1 - Y-X Turn enabled for associated Source-Target Pair, 0 - Y-X Turn disabled	RW	1'b0
32	xy_override_enable_{(2*index)+1}	X-Y Route Override Enable: 1 - X-Y Route override enabled for associated Source-Target Pair, 0 - X-Y Route override disabled	RW	1'b0

The following image shows the lower register bit assignments.

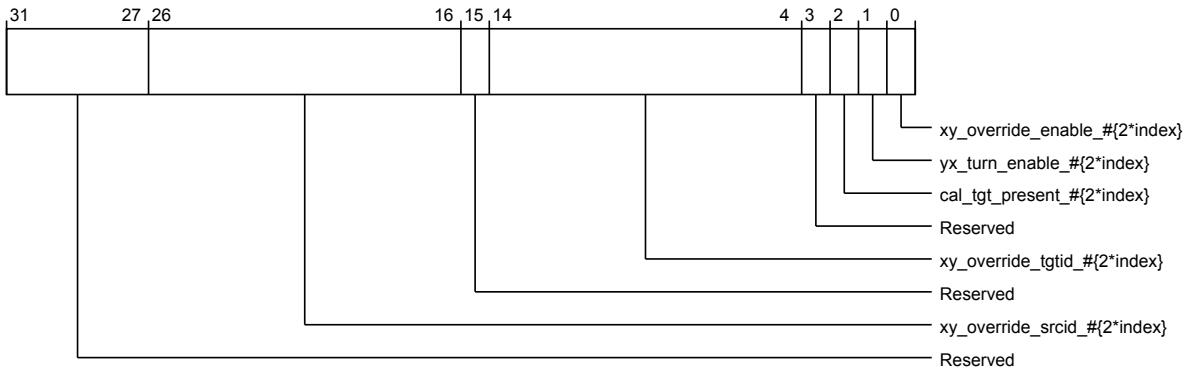


Figure 3-1172 por_mxp_por_mxp_xy_override_sel_0-7 (low)

The following table shows the por_mxp_xy_override_sel_0-7 lower register bit assignments.

Table 3-1192 por_mxp_por_mxp_xy_override_sel_0-7 (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:16	xy_override_srcid_{2*index}	11-bit Source ID associated with the XY Override. This field is used in the LUP to determine if XY route for the associated source-target pair needs to be overridden.	RW	11'b0
15	Reserved	Reserved	RO	-
14:4	xy_override_tgtid_{2*index}	11-bit Target ID associated with the XY Override. This field is used in the LUP to determine if XY route for the associated source-target pair needs to be overridden.	RW	11'b0
3	Reserved	Reserved	RO	-
2	cal_tgt_present_{2*index}	CAL TGT Presence Indication for XY Route Override of all devices behind CAL: 1 - CAL4 TGT Present for XY Route Override of all devices behind CAL, 0 - CAL2 TGT or no CAL TGT Present for XY Route Override of all devices behind CAL.	RW	1'b0
1	yx_turn_enable_{2*index}	Y-X Turn Enable: 1 - Y-X Turn enabled for associated Source-Target Pair, 0 - Y-X Turn disabled	RW	1'b0
0	xy_override_enable_{2*index}	X-Y Route Override Enable: 1 - X-Y Route override enabled for associated Source-Target Pair, 0 - X-Y Route override disabled	RW	1'b0

por_mxp_p_0-5_pa2setaddr_slc

This register repeats 5 times. It parametrized by the index from 0 to 5. Functions as the control register of PA to SetAddr and vice versa conversion for HNF-SLC on XP port #*{index}*. NOTE: There will be max. of 6 MXP Port registers based on MXP_NUM_DEV_PORT_PARAM value. Each successive MXP Port register will be at the next 8 byte address boundary.

Its characteristics are:

Type	RW

Register width (Bits) 64

Address offset 16'hCD0 + #{32*[0, 1, 2, 3, 4, 5]}

Register reset 64'b0110

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_mxp_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.



Figure 3-1173 por_mxp_por_mxp_p_0-5_pa2setaddr_slc (high)

The following table shows the por_mxp_p_0-5_pa2setaddr_slc higher register bit assignments.

Table 3-1193 por_mxp_por_mxp_p_0-5_pa2setaddr_slc (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

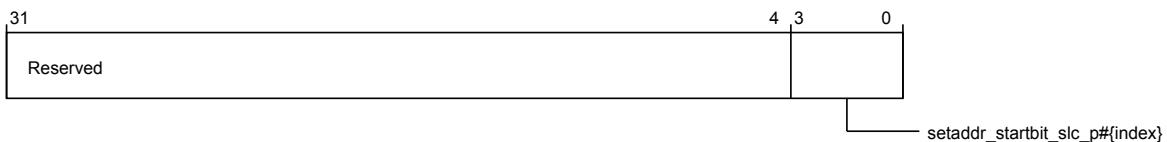


Figure 3-1174 por_mxp_por_mxp_p_0-5_pa2setaddr_slc (low)

The following table shows the por_mxp_p_0-5_pa2setaddr_slc lower register bit assignments.

Table 3-1194 por_mxp_por_mxp_p_0-5_pa2setaddr_slc (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	setaddr_startbit_slc_p#{index}	SLC: SetAddr starting bit for SLC in HNF connected to port p#{index} 4'b0110: Setaddr starts from PA[6] 4'b0111: Setaddr starts from PA[7] 4'b1000: Setaddr starts from PA[8] 4'b1001: Setaddr starts from PA[9] 4'b1010: Setaddr starts from PA[10] 4'b1011: Setaddr starts from PA[11] 4'b1100: Setaddr starts from PA[12]	RW	4'b0110

por_mxp_p_0-5_pa2setaddr_sf

This register repeats 5 times. It parametrized by the index from 0 to 5. Functions as the control register of PA to Set/TagAddr and vice versa conversion for HNF-SF on XP port #*{index}*. NOTE: There will be max. of 6 MXP Port registers based on MXP_NUM_DEV_PORT_PARAM value. Each successive MXP Port register will be at the next 8 byte address boundary.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hCD8 + # <i>{32*[0, 1, 2, 3, 4, 5]}</i>
Register reset	64'b0110
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mxp_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.



Figure 3-1175 por_mxp_por_mxp_p_0-5_pa2setaddr_sf (high)

The following table shows the por_mxp_p_0-5_pa2setaddr_sf higher register bit assignments.

Table 3-1195 por_mxp_por_mxp_p_0-5_pa2setaddr_sf (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

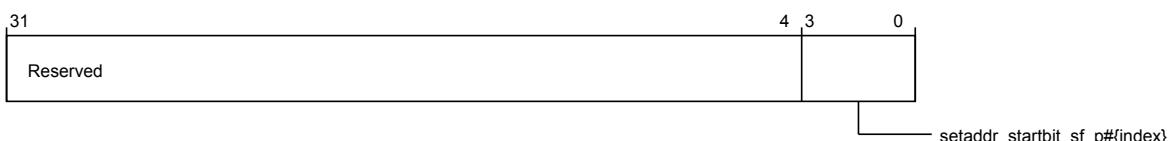


Figure 3-1176 por_mxp_por_mxp_p_0-5_pa2setaddr_sf (low)

The following table shows the por_mxp_p_0-5_pa2setaddr_sf lower register bit assignments.

Table 3-1196 por_mxp_por_mxp_p_0-5_pa2setaddr_sf (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	setaddr_startbit_sf_p#{index}	SF: SetAddr starting bit for SF in HNF connected to port p#{index} 4'b0110: Setaddr starts from PA[6] 4'b0111: Setaddr starts from PA[7] 4'b1000: Setaddr starts from PA[8] 4'b1001: Setaddr starts from PA[9] 4'b1010: Setaddr starts from PA[10] 4'b1011: Setaddr starts from PA[11] 4'b1100: Setaddr starts from PA[12]	RW	4'b0110

por_mxp_p_0-5_pa2setaddr_flex_slc

This register repeats 5 times. It parametrized by the index from 0 to 5. Functions as the SLC control register of PA to Set/TagAddr and vice versa conversion for HNF (flexible) on XP port #{index}. NOTE: There will be max. of 6 MXP Port registers based on MXP_NUM_DEV_PORT_PARAM value. Each successive MXP Port register will be at the next 8 byte address boundary.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hCE0 + #{32*[0, 1, 2, 3, 4, 5]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mxp_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.

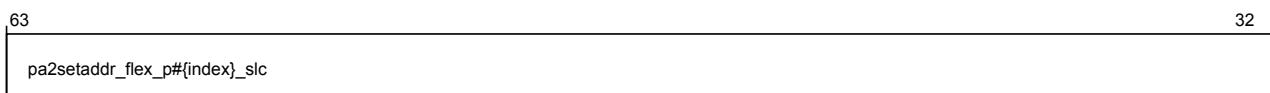


Figure 3-1177 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_slc (high)

The following table shows the por_mxp_p_0-5_pa2setaddr_flex_slc higher register bit assignments.

Table 3-1197 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_slc (high)

Bits	Field name	Description	Type	Reset
63:32	pa2setaddr_flex_p#{index}_slc	FLEXIBLE: PA to SET/TAG ADDR and vice versa conversion config field for HNF connected to port p#{index}	RW	64'b0

The following image shows the lower register bit assignments.

31	pa2setaddr_flex_p#{index}_slc	0
----	-------------------------------	---

Figure 3-1178 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_slc (low)

The following table shows the por_mxp_p_0-5_pa2setaddr_flex_slc lower register bit assignments.

Table 3-1198 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_slc (low)

Bits	Field name	Description	Type	Reset
31:0	pa2setaddr_flex_p#{index}_slc	FLEXIBLE: PA to SET/TAG ADDR and vice versa conversion config field for HNF connected to port p#{index}	RW	64'b0

por_mxp_p_0-5_pa2setaddr_flex_sf

This register repeats 5 times. It parametrized by the index from 0 to 5. Functions as the SF control register of PA to Set/TagAddr and vice versa conversion for HNF (flexible) on XP port #{index}. NOTE: There will be max. of 6 MXP Port registers based on MXP_NUM_DEV_PORT_PARAM value. Each successive MXP Port register will be at the next 8 byte address boundary.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hCE8 + #{32*[0, 1, 2, 3, 4, 5]}
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_mxp_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.

63	pa2setaddr_flex_p#{index}_sf	32
----	------------------------------	----

Figure 3-1179 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_sf (high)

The following table shows the por_mxp_p_0-5_pa2setaddr_flex_sf higher register bit assignments.

Table 3-1199 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_sf (high)

Bits	Field name	Description	Type	Reset
63:32	pa2setaddr_flex_p#{index}_sf	FLEXIBLE: PA to SET/TAG ADDR conversion and vice versa config field for HNF connected to port p#{index}	RW	64'b0

The following image shows the lower register bit assignments.



Figure 3-1180 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_sf (low)

The following table shows the por_mxp_p_0-5_pa2setaddr_flex_sf lower register bit assignments.

Table 3-1200 por_mxp_por_mxp_p_0-5_pa2setaddr_flex_sf (low)

Bits	Field name	Description	Type	Reset
31:0	pa2setaddr_flex_p#{index}_sf	FLEXIBLE: PA to SET/TAG ADDR conversion and vice versa config field for HNF connected to port p#{index}	RW	64'b0

3.3.19 HN-F register descriptions

This section lists the HN-F registers.

por_hnf_node_info

Provides component identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h0

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

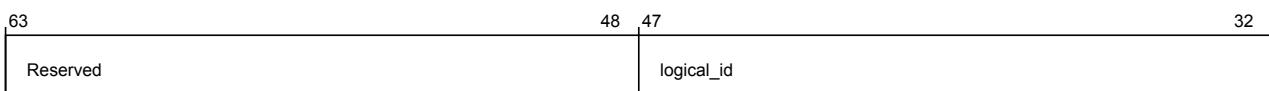


Figure 3-1181 por_hnf_por_hnf_node_info (high)

The following table shows the por_hnf_node_info higher register bit assignments.

Table 3-1201 por_hnf_por_hnf_node_info (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47:32	logical_id	Component logical ID	RO	Configuration dependent

The following image shows the lower register bit assignments.

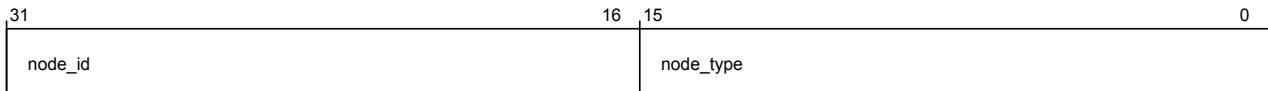


Figure 3-1182 por_hnf_por_hnf_node_info (low)

The following table shows the por_hnf_node_info lower register bit assignments.

Table 3-1202 por_hnf_por_hnf_node_info (low)

Bits	Field name	Description	Type	Reset
31:16	node_id	Component node ID	RO	Configuration dependent
15:0	node_type	CMN-600 node type identifier	RO	16'h0005

por_hnf_child_info

Provides component child identification information.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h80

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

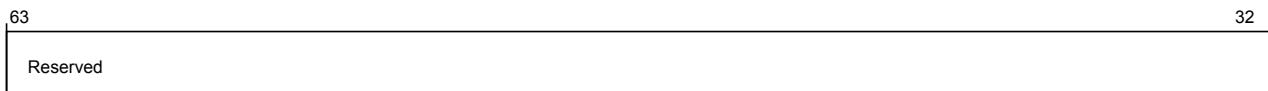


Figure 3-1183 por_hnf_por_hnf_child_info (high)

The following table shows the por_hnf_child_info higher register bit assignments.

Table 3-1203 por_hnf_por_hnf_child_info (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

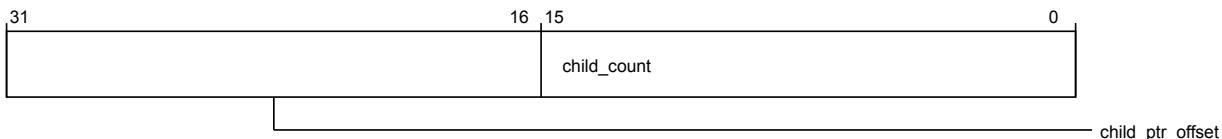


Figure 3-1184 por_hnf_por_hnf_child_info (low)

The following table shows the por_hnf_child_info lower register bit assignments.

Table 3-1204 por_hnf_por_hnf_child_info (low)

Bits	Field name	Description	Type	Reset
31:16	child_ptr_offset	Starting register offset which contains pointers to the child nodes	RO	16'h0
15:0	child_count	Number of child nodes; used in discovery process	RO	16'b0

por_hnf_secure_register_groups_override

Allows non-secure access to predefined groups of secure registers.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h980

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

The following image shows the higher register bit assignments.



Figure 3-1185 por_hnf_por_hnf_secure_register_groups_override (high)

The following table shows the por_hnf_secure_register_groups_override higher register bit assignments.

Table 3-1205 por_hnf_por_hnf_secure_register_groups_override (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

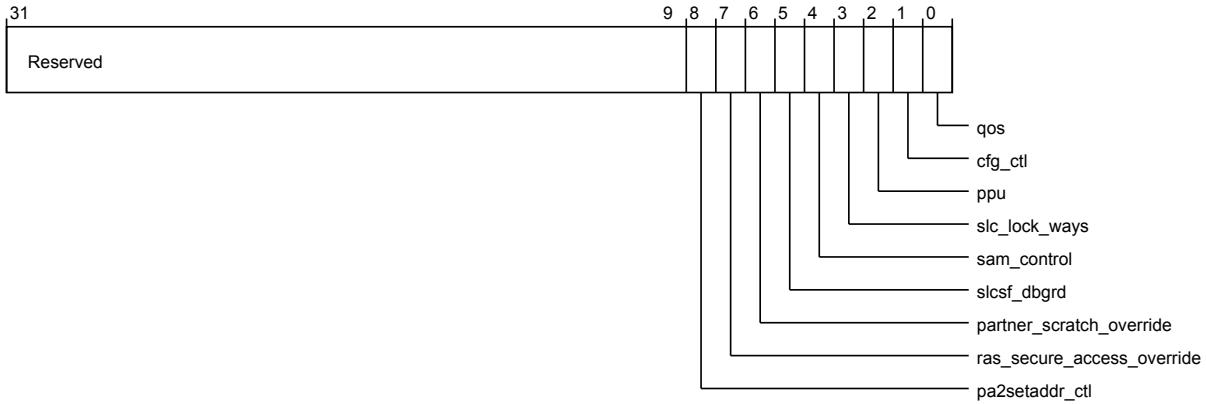


Figure 3-1186 por_hnf_por_hnf_secure_register_groups_override (low)

The following table shows the por_hnf_secure_register_groups_override lower register bit assignments.

Table 3-1206 por_hnf_por_hnf_secure_register_groups_override (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	pa2setaddr_ctl	Allow non-secure access to secure PA2SETADDR registers	RW	1'b0
7	ras_secure_access_override	Allow non-secure access to secure RAS registers	RW	1'b0
6	partner_scratch_override	Allows non-secure access to secure Partenr scratch registers	RW	1'b0
5	slesf_dbgrd	Allows non-secure access to secure SLC/SF debug read registers	RW	1'b0
4	sam_control	Allows non-secure access to secure HN-F SAM control registers	RW	1'b0
3	slc_lock_ways	Allows non-secure access to secure cache way locking registers	RW	1'b0
2	ppu	Allows non-secure access to secure power policy registers	RW	1'b0
1	cfg_ctl	Allows non-secure access to secure configuration control register (por_hnf_cfg_ctl)	RW	1'b0
0	qos	Allows non-secure access to secure QoS registers	RW	1'b0

por_hnf_unit_info

Provides component identification information for HN-F.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h900

Register reset Configuration dependent

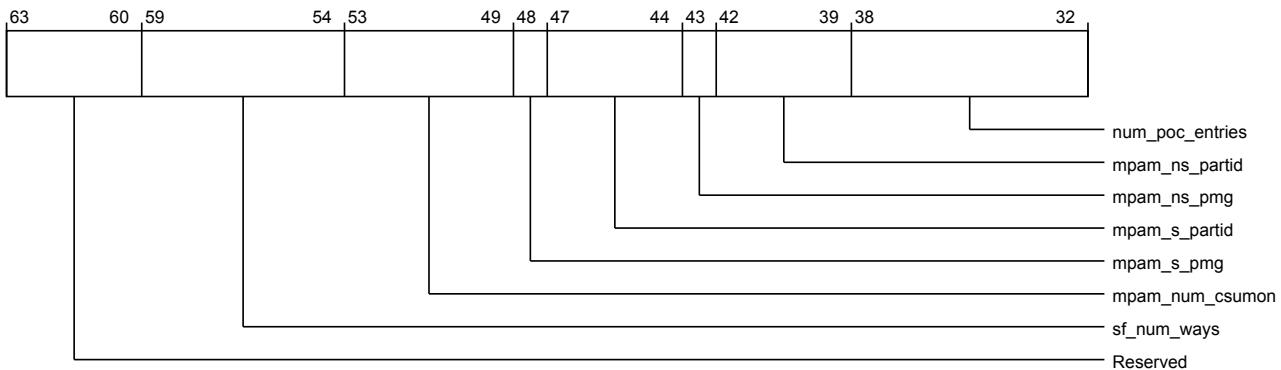


Figure 3-1187 por_hnf_por_hnf_unit_info (high)

The following table shows the por_hnf_unit_info higher register bit assignments.

Table 3-1207 por_hnf_por_hnf_unit_info (high)

Bits	Field name	Description	Type	Reset
63:60	Reserved	Reserved	RO	-
59:54	sf_num_ways	Number of cache ways in the SF	RO	-
53:49	mpam_num_csumon	Number of Cache Storage Usage Monitors for MPAM	RO	Configuration dependent
48	mpam_s_pmg	MPAM Secure supported PMGs 1'b0: 1 S PMG 1'b1: 2 S PMG	RO	-
47:44	mpam_s_partid	MPAM Secure supported PARTIDs 4'b0000: 1 S PARTID 4'b0001: 2 S PARTID 4'b0010: 4 S PARTID 4'b0011: 8 S PARTID 4'b0100: 16 S PARTID 4'b0101: 32 S PARTID 4'b0110: 64 S PARTID 4'b0111: 128 S PARTID 4'b1000: 256 S PARTID 4'b1001: 512 S PARTID	RO	-
43	mpam_ns_pmg	MPAM Non-Secure supported PMGs 1'b0: 1 NS PMG 1'b1: 2 NS PMG	RO	-

Table 3-1207 por_hnf_por_hnf_unit_info (high) (continued)

Bits	Field name	Description	Type	Reset
42:39	mpam_ns_partid	MPAM Non-Secure supported PARTIDs 4'b0000: 1 NS PARTID 4'b0001: 2 NS PARTID 4'b0010: 4 NS PARTID 4'b0011: 8 NS PARTID 4'b0100: 16 NS PARTID 4'b0101: 32 NS PARTID 4'b0110: 64 NS PARTID 4'b0111: 128 NS PARTID 4'b1000: 256 NS PARTID 4'b1001: 512 NS PARTID	RO	-
38:32	num_poc_entries	Number of POCQ entries	RO	Configuration dependent

The following image shows the lower register bit assignments.

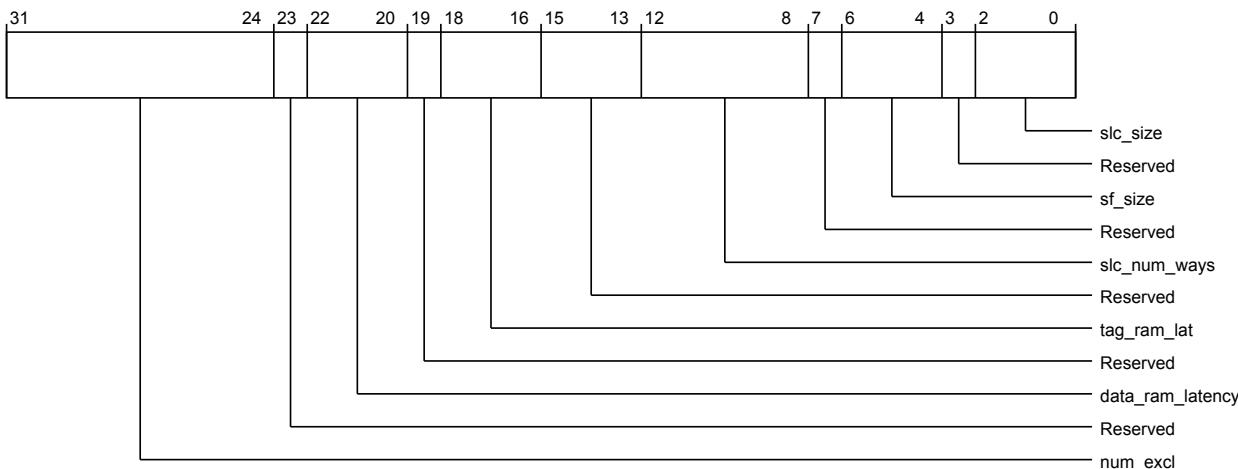


Figure 3-1188 por_hnf_por_hnf_unit_info (low)

The following table shows the por_hnf_unit_info lower register bit assignments.

Table 3-1208 por_hnf_por_hnf_unit_info (low)

Bits	Field name	Description	Type	Reset
31:24	num_excl	Number of exclusive monitors	RO	-
23	Reserved	Reserved	RO	-
22:20	data_ram_latency	SLC data RAM latency (in cycles)	RO	-
19	Reserved	Reserved	RO	-

Table 3-1208 por_hnf_por_hnf_unit_info (low) (continued)

Bits	Field name	Description	Type	Reset
18:16	tag_ram_lat	SLC tag RAM latency (in cycles)	RO	-
15:13	Reserved	Reserved	RO	-
12:8	slc_num_ways	Number of cache ways in the SLC	RO	-
7	Reserved	Reserved	RO	-
6:4	sf_size	SF size 3'b000: 512KB 3'b001: 1MB 3'b010: 2MB 3'b011: 4MB 3'b100: 8MB	RO	-
3	Reserved	Reserved	RO	-
2:0	slc_size	SLC size 3'b000: No SLC 3'b001: 128KB 3'b010: 256KB 3'b011: 512KB 3'b100: 1MB 3'b101: 2MB 3'b110: 3MB 3'b111: 4MB	RO	-

por_hnf_unit_info_1

Provides component identification information for HN-F.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h908

Register reset Configuration dependent

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

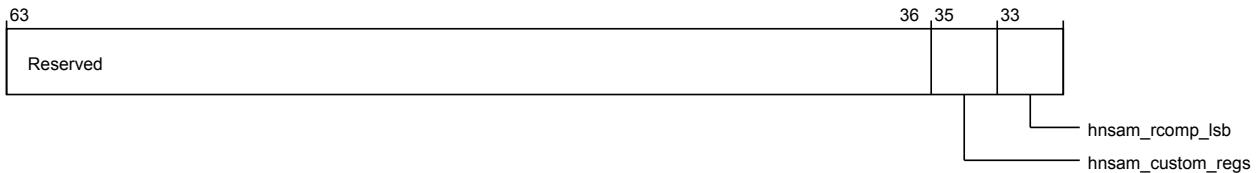


Figure 3-1189 por_hnf_por_hnf_unit_info_1 (high)

The following table shows the por_hnf_unit_info_1 higher register bit assignments.

Table 3-1209 por_hnf_por_hnf_unit_info_1 (high)

Bits	Field name	Description	Type	Reset
63:36	Reserved	Reserved	RO	-
35:34	hnsam_custom_regs	Number of customer specific registers for customer implemented logic	RO	Configuration dependent
33:32	hnsam_rcomp_lsb	Defines the minimum size of HTG when POR_HNSAM_RCOMP_EN_PARAM = 1, 20 value defines minimum size as 1MB and 26 value defines minimum size as 64MB	RO	Configuration dependent

The following image shows the lower register bit assignments.

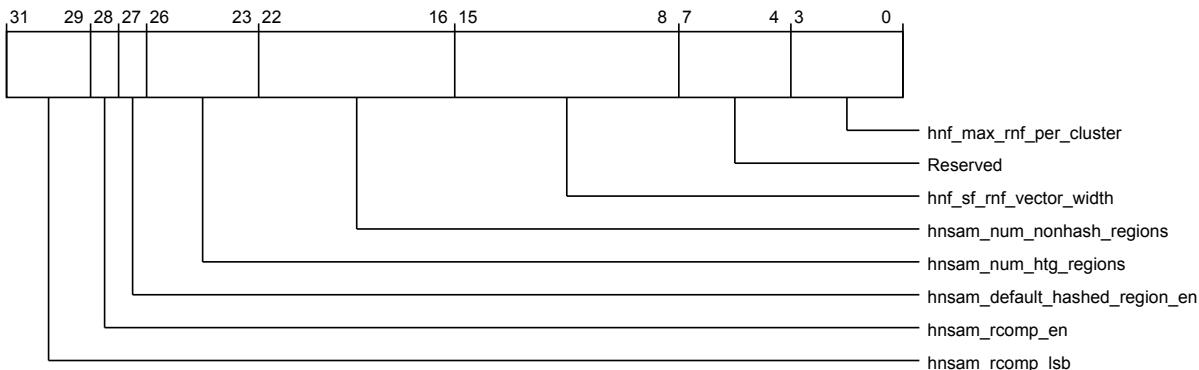


Figure 3-1190 por_hnf_por_hnf_unit_info_1 (low)

The following table shows the por_hnf_unit_info_1 lower register bit assignments.

Table 3-1210 por_hnf_por_hnf_unit_info_1 (low)

Bits	Field name	Description	Type	Reset
31:29	hnsam_rcomp_lsb	Defines the minimum size of HTG when POR_HNSAM_RCOMP_EN_PARAM = 1, 20 value defines minimum size as 1MB and 26 value defines minimum size as 64MB	RO	Configuration dependent
28	hnsam_rcomp_en	Enable Range based address comparison for HNSAM HTG/ Nonhashed groups. Program start address and end address	RO	Configuration dependent
27	hnsam_default_hashed_region_en	Enable default hashed group for HNSAM. To support backward compatible, set this parameter	RO	Configuration dependent

Table 3-1210 por_hnf_por_hnf_unit_info_1 (low) (continued)

Bits	Field name	Description	Type	Reset
26:23	hnsam_num_htg_regions	Number of HTG regions supported by the HNSAM	RO	Configuration dependent
22:16	hnsam_num_nonhash_regions	Number of non-hashed regions supported by the HNSAM	RO	Configuration dependent
15:8	hnf_sf_rnf_vector_width	Total Number of bits in RNF tracking vector in the Snoop Filter (Total SF_VEC_WIDTH = (TOTAL_RNF/ HNF_MAX_CLUSTER_PARAM) +HNF_SF_ADD_VECTOR_WIDTH)	RO	Configuration dependent
7:4	Reserved	Reserved	RO	-
3:0	hnf_max_rnf_per_cluster	Describes the maximum number of RN-F's in a single cluster	RO	Configuration dependent

por_hnf_cfg_ctl

Functions as the configuration control register for HN-F.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA00

Register reset 64'b0010000000000000

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

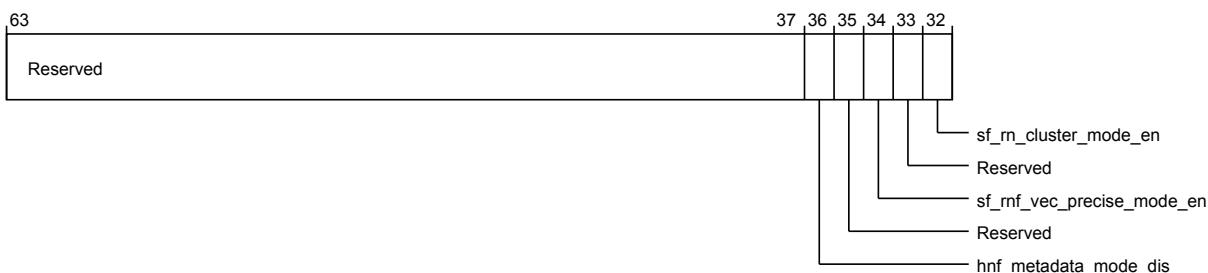


Figure 3-1191 por_hnf_por_hnf_cfg_ctl (high)

The following table shows the por_hnf_cfg_ctl higher register bit assignments.

Table 3-1211 por_hnf_por_hnf_cfg_ctl (high)

Bits	Field name	Description	Type	Reset
63:37	Reserved	Reserved	RO	-
36	hnf_metadata_mode_dis	Disables the METADATA features in HNF when set to 1'b1	RW	1'b0
35	Reserved	Reserved	RO	-
34	sf_rnf_vec_precise_mode_en	Enables the snoop filter's precise RNF vector in clustered mode	RW	1'b0
33	Reserved	Reserved	RO	-
32	sf_rn_cluster_mode_en	Enables the snoop filter clustering of the RN-F ID's using programmable registers	RW	1'b1

The following image shows the lower register bit assignments.

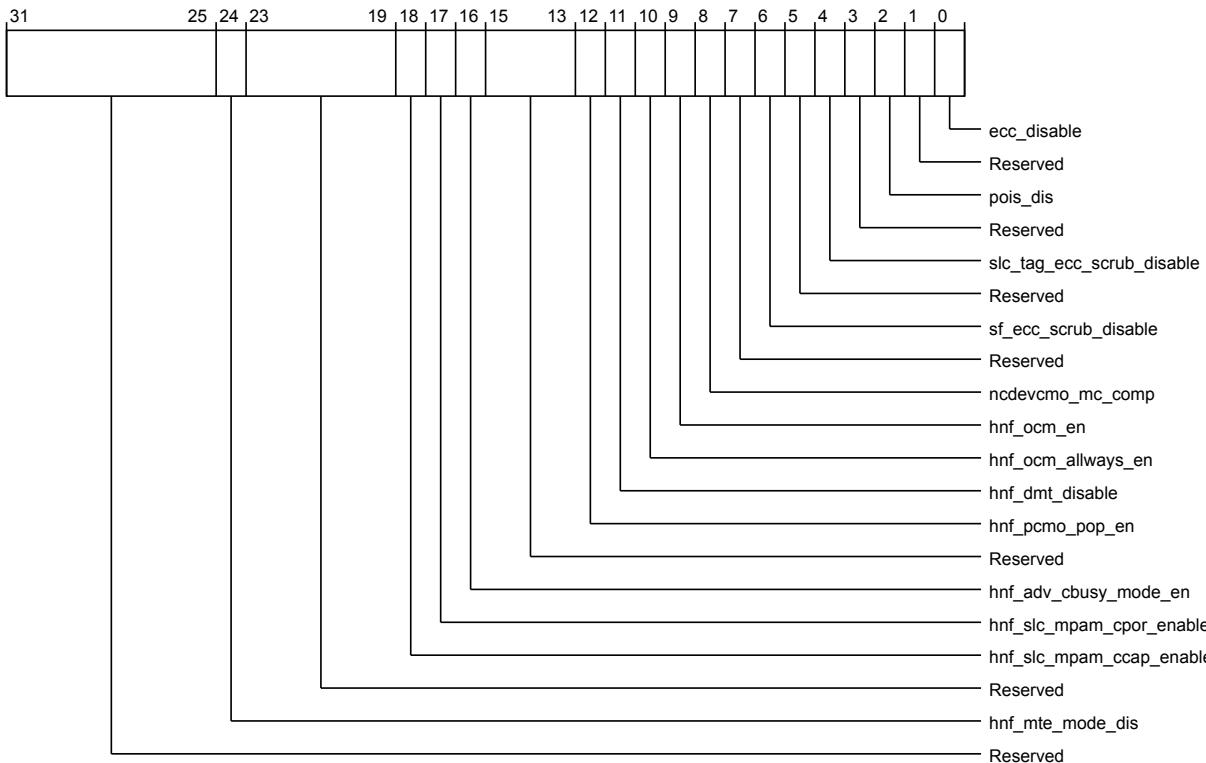


Figure 3-1192 por_hnf_por_hnf_cfg_ctl (low)

The following table shows the por_hnf_cfg_ctl lower register bit assignments.

Table 3-1212 por_hnf_por_hnf_cfg_ctl (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	hnf_mte_mode_dis	Disables the MTE features in HNF when set to 1'b1	RW	1'b0
23:19	Reserved	Reserved	RO	-

Table 3-1212 por_hnf_por_hnf_cfg_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
18	hnf_slc_mpam_ccap_enable	Enable MPAM Cache Capacity Partitioning for SLC 1'b1: Cache Capacity Partitioning is enabled if supported in Hardware. 1'b0: Cache Capacity Partitioning is disabled for SLC. NOTE: If MPAM is disabled at build time, this bit has no meaning.	RW	1'b0
17	hnf_slc_mpam_cpor_enable	Enable MPAM Cache Portion Partitioning for SLC 1'b1: Cache Portion Partitioning is enabled if supported in Hardware. 1'b0: Cache Portion Partitioning is disabled for SLC. NOTE: If MPAM is disabled at build time, this bit has no meaning.	RW	1'b0
16	hnf_adv_cbusy_mode_en	Enables the advanced features of HNF CBusy handling	RW	1'b0
15:13	Reserved	Reserved	RO	-
12	hnf_pcmo_pop_en	Terminates PCMO in HNF when this bit is set to 1'b1	RW	1'b0
11	hnf_dmt_disable	Disables DMT when set	RW	1'b0
10	hnf_ocm_allways_en	Enables all SLC ways with OCM	RW	1'b0
9	hnf_ocm_en	Enables region locking with OCM support	RW	1'b0
8	ncdevcmo_mc_comp	Disables HN-F completion when set NOTE: When set, HN-F sends completion for the following transactions received after completion from SN: 1. Non-cacheable WriteNoSnp 2. Device WriteNoSnp 3. CMO (cache maintenance operations) CONSTRAINT: When this bit is set, por_mni_cfg_ctl.dis_ncwr_stream and por_rnd_cfg_ctl.dis_ncwr_stream must also be set.	RW	1'b0
7	Reserved	Reserved	RO	-
6	sf_ecc_scrub_disable	Disables SF tag single-bit ECC error scrubbing when set	RW	1'b0
5	Reserved	Reserved	RO	-
4	slc_tag_ecc_scrub_disable	Disables SLC tag single-bit ECC error scrubbing when set	RW	1'b0
3	Reserved	Reserved	RO	-
2	pois_dis	Disables parity error data poison when set	RW	1'b0
1	Reserved	Reserved	RO	-
0	ecc_disable	Disables SLC and SF ECC generation/detection when set	RW	1'b0

por_hnf_aux_ctl

Functions as the auxiliary control register for HN-F.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA08
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

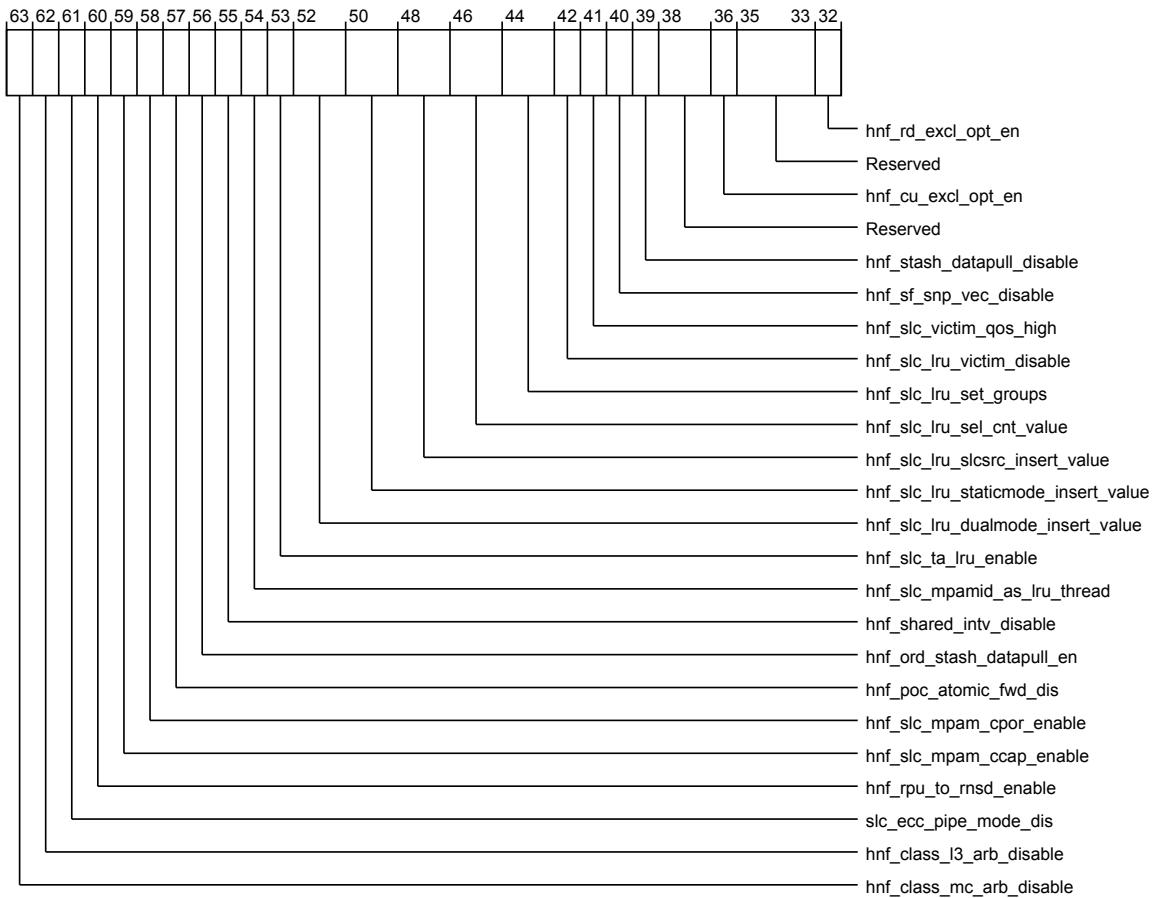


Figure 3-1193 por_hnf_por_hnf_aux_ctl (high)

The following table shows the por_hnf_aux_ctl higher register bit assignments.

Table 3-1213 por_hnf_por_hnf_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63	hnf_class_mc_arb_disable	Disables Class based arbitration for MC Arbitration. 1'b0: Use Class based arbitration for MC Pipeline. 1'b1: Use QoS based arbitration for MC Pipeline. Same as previous generation of QoS.	RW	1'b0
62	hnf_class_l3_arb_disable	Disables Class based arbitration for L3 Arbitration. 1'b0: Use Class based arbitration for L3 Pipeline. 1'b1: Use QoS based arbitration for L3 Pipeline. Same as previous generation of QoS.	RW	1'b0
61	slc_ecc_pipe_mode_dis	Disables inline ECC pipe mode in SLC. CONSTRAINT: Must be programmed at boot time.	RW	1'b1
60	hnf_rpu_to_rnsd_enable	Enables HN-F to treat ReadPrefUnique ops as ReadNotSharedDirty	RW	1'b0
59	hnf_slc_mpam_ccap_enable	Enable MPAM Cache Capacity Partitioning for SLC 1'b1: Cache Capacity Partitioning is enabled if supported in Hardware. 1'b0: Cache Capacity Partitioning is disabled for SLC. NOTE: This bit moved to cfg_ctl and will be removed in future version of ICN. NOTE: If MPAM is disabled at build time, this bit has no meaning.	RW	1'b0
58	hnf_slc_mpam_cpor_enable	Enable MPAM Cache Portion Partitioning for SLC 1'b1: Cache Portion Partitioning is enabled if supported in Hardware. 1'b0: Cache Portion Partitioning is disabled for SLC. NOTE: This bit moved to cfg_ctl and will be removed in future version of ICN. NOTE: If MPAM is disabled at build time, this bit has no meaning.	RW	1'b0
57	hnf_poc_atomic_fwd_dis	Disable the atomic data forwarding in POCQ	RW	1'b0
56	hnf_ord_stash_datapull_en	Enables stash datapull for ordered write stash requests	RW	1'b1
55	hnf_shared_intv_disable	Disables snoop requests to CHIB RN-F with shared copy	RW	Configuration dependent

Table 3-1213 por_hnf_por_hnf_aux_ctl (high) (continued)

Bits	Field name	Description	Type	Reset
54	hnf_slc_mpamid_as_lru_thread	<p>Use MPAM PARTID as ThreadID for Thread Aware eLRU</p> <p>1'b0: ThreadID is based on LPID+LID for Thread Aware eLRU.</p> <p>1'b1: ThreadID is based on MPAM PARTID+NS for Thread Aware eLRU.</p> <p>Note: MPAM PARTID is used only if MPAM is enabled.</p>	RW	1'b0
53	hnf_slc_ta_lru_enable	<p>Thread Aware eLRU enable</p> <p>1'b0: ThreadID used for eLRU is zero.</p> <p>1'b1: ThreadID used for eLRU is based on MPAMID or LPID +LID.</p> <p>Note: If SLC size is less than 256KB, this bit is ignore.</p>	RW	1'b0
52:51	hnf_slc_lru_dualmode_insert_value	<p>Insertion value for Dual mode eLRU</p> <p>NOTE: Default is 2'b11.</p>	RW	2'b11
50:49	hnf_slc_lru_staticmode_insert_value	<p>Insertion value for Static mode eLRU</p> <p>NOTE: Default is 2'b10.</p>	RW	2'b10
48:47	hnf_slc_lru_slcsrc_insert_value	<p>Insertion value if SLC source bit is set</p> <p>NOTE: Default is 2'b00.</p>	RW	2'b00
46:45	hnf_slc_lru_sel_cnt_value	<p>Selection counter value for eLRU to determine which group policy is more effective</p> <p>2'b00: Sel counter is like an 8-bit range; upper limit is 255; middle point is 128</p> <p>2'b01: Sel counter is like a 9-bit range; upper limit is 511; middle point is 256</p> <p>2'b10: Sel counter is like a 10-bit range; upper limit is 1023; middle point is 512</p> <p>2'b11: Sel counter is like an 11-bit range; upper limit is 2047; middle point is 1024</p> <p>NOTE: Default is 10-bit with counter reset to a value of 512.</p>	RW	2'b10
44:43	hnf_slc_lru_set_groups	<p>Number of sets in monitor group for enhance LRU</p> <p>2'b00: 16</p> <p>2'b01: 32</p> <p>2'b10: 64</p> <p>2'b11: 128</p> <p>NOTE: Default is 32 sets per monitor group. If cache size is small (128KB or less), there would be only one set per group.</p>	RW	2'b01

Table 3-1213 por_hnf_por_hnf_aux_ctl (high) (continued)

Bits	Field name	Description	Type	Reset
42	hnf_slc_lru_victim_disable	Disable enhanced LRU based victim selection for SLC 1'b0: SLC victim selection is based on eLRU. 1'b1: SLC victim selection is based on LFSR. NOTE: Victim selection for SF is always LFSR-based.	RW	1'b1
41	hnf_slc_victim_qos_high	SLC victim QoS behavior for SN write request 1'b0: Each victim inherits the QoS value of the request which caused it 1'b1: All victims use high QoS class (14)	RW	1'b0
40	hnf_sf_snp_vec_disable	Disables SF snoop vector when set	RW	1'b0
39	hnf_stash_datapull_disable	Disables HN-F stash data pull support when set	RW	1'b0
38:37	Reserved	Reserved	RO	-
36	hnf_cu_excl_opt_en	CleanUnique exclusive optimization enable	RW	1'b1
35:33	Reserved	Reserved	RO	-
32	hnf_rd_excl_opt_en	ReadNotSharedDirty exclusive optimization enable	RW	1'b0

The following image shows the lower register bit assignments.

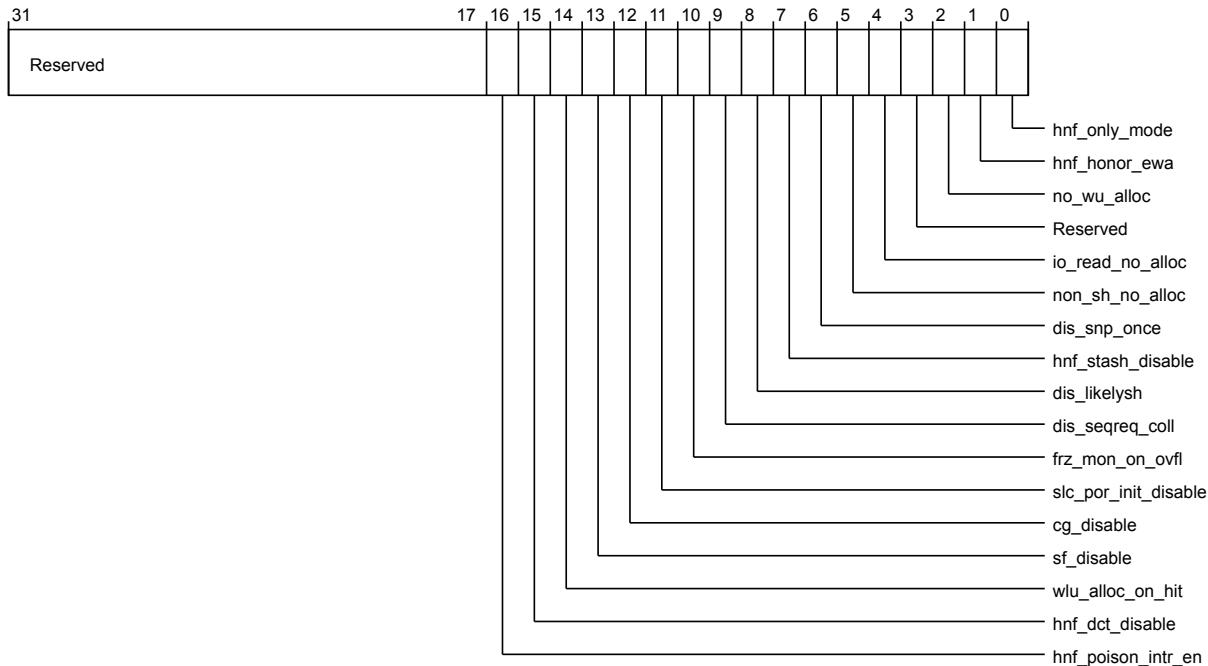


Figure 3-1194 por_hnf_por_hnf_aux_ctl (low)

The following table shows the por_hnf_aux_ctl lower register bit assignments.

Table 3-1214 por_hnf_por_hnf_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:17	Reserved	Reserved	RO	-
16	hnf_poison_intr_en	Enables reporting an interrupt by HN-F when poison is detected at SLC	RW	Configuration dependent
15	hnf_dct_disable	Disables DCT when set	RW	Configuration dependent
14	wlu_alloc_on_hit	Forces WLU requests to allocate if the line hit in SLC	RW	1'b0
13	sf_disable	Disables SF	RW	1'b0
12	cg_disable	Disables HN-F architectural clock gates	RW	1'b0
11	slc_por_init_disable	Disables SLC and SF initialization on Reset	RW	1'b0
10	frz_mon_on_ovfl	Freezes the exclusive monitors	RW	1'b0
9	dis_seqreq_coll		RW	1'b0
8	dis_likelysh	Disables Likely Shared based allocations	RW	1'b0
7	hnf_stash_disable	Disables HN-F stash support	RW	Configuration dependent
6	dis_snp_once	When set, disables SnpOnce and converts to SnpShared	RW	Configuration dependent
5	non_sh_no_alloc	Disables SLC allocation for non-shareable cacheable transactions when set	RW	1'b0
4	io_read_no_alloc	When set, disables ReadOnce and ReadNoSnp allocation in SLC from RN-Is	RW	1'b0
3	Reserved	Reserved	RO	-
2	no_wu_alloc	Disables WriteUnique/WriteLineUnique allocations in SLC when set	RW	1'b0
1	hnf_honor_ewa	When set, postpones completion for writes where EWA=0 in the request until HN-F receives completion from MC or SBSX	RW	1'b1
0	hnf_only_mode	Enables HN-F only mode; disables SLC and SF when set	RW	1'b0

por_hnf_r2_aux_ctl

Functions as the auxiliary control register for HN-F for CMN-600 R2 features.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hA10

Register reset 64'b100000000011010000

Usage constraints Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-1195 por_hnf_por_hnf_r2_aux_ctl (high)

The following table shows the por_hnf_r2_aux_ctl higher register bit assignments.

Table 3-1215 por_hnf_por_hnf_r2_aux_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

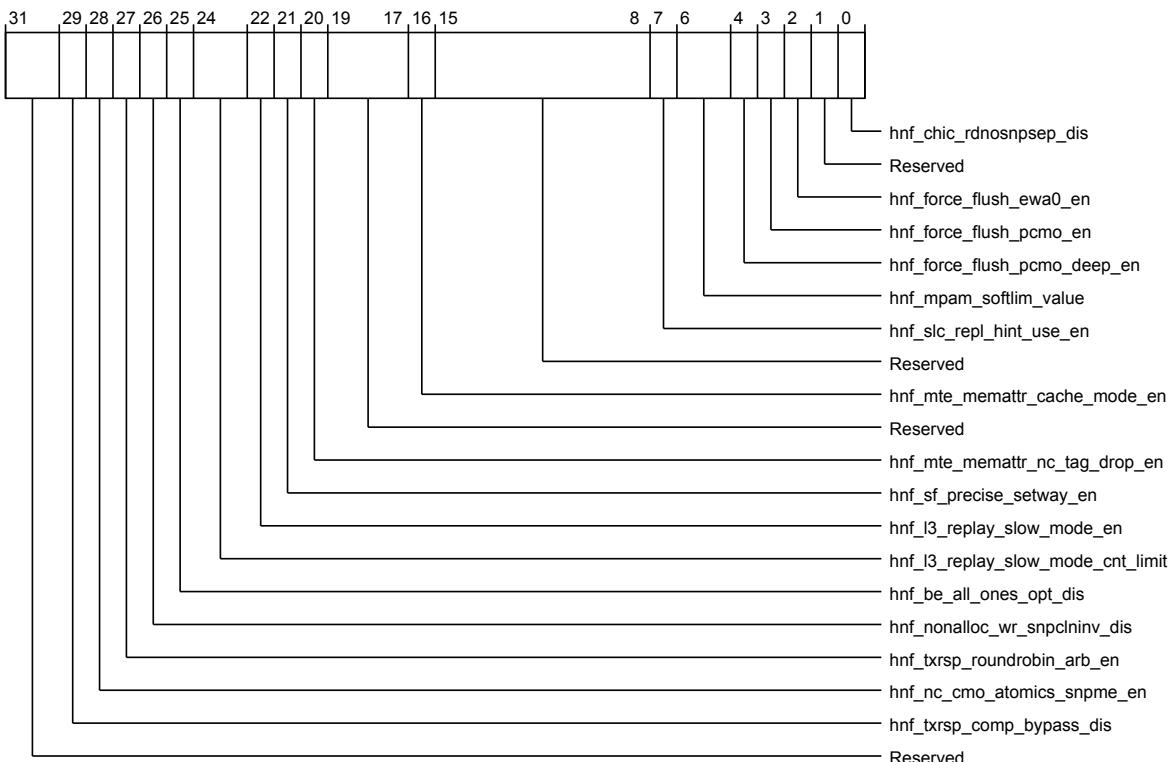


Figure 3-1196 por_hnf_por_hnf_r2_aux_ctl (low)

The following table shows the por_hnf_r2_aux_ctl lower register bit assignments.

Table 3-1216 por_hnf_por_hnf_r2_aux_ctl (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29	hnf_txrsp_comp_bypass_dis	When set, TXRSP COMP bypass gets disabled for WEOE/EVICT	RW	1'b1
28	hnf_nc_cmo_atomics_snpme_en	When set to 1, all incoming non-cachable atomics and cmo's from RNF will be back-snooped	RW	1'b0

Table 3-1216 por_hnf_por_hnf_r2_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
27	hnf_txrsp_roundrobin_arb_en	When set, TXRSP final arb is changed from Weighted roundrobin to roundrobin	RW	1'b0
26	hnf_nonalloc_wr_snpclminv_dis	Disable the.snp type of.snp_cln_inv on non-allocating writes. Send.snp_uniq instead	RW	1'b0
25	hnf_be_all_ones_opt_dis	Disable the optimizations related to BE=1's hint on WR_PTL from RNI	RW	1'b0
24:23	hnf_l3_replay_slow_mode_cnt_limit	L3 arbitration throttle count limit, when enabled. 00: L3 Throttle is enabled after 512 setway haz replays 01: L3 Throttle is enabled after 1024 setway haz replays 10: L3 Throttle is enabled after 2048 setway haz replays 11: L3 Throttle is enabled after 4096 setway haz replays	RW	2'b00
22	hnf_l3_replay_slow_mode_en	Enables L3 arbitration slow mode in case of constant replays, when set to 1'b1	RW	1'b0
21	hnf_sf_precise_setway_en	Enables Precise setway hazard, when set to 1'b1	RW	1'b0
20	hnf_mte_memattr_nc_tag_drop_en	Enables HNF to drop any dirty tags for Non-Cacheable memory, when set to 1'b1	RW	1'b0
19:17	Reserved	Reserved	RO	-
16	hnf_mte_memattr_cache_mode_en	When set to 1'b1, it enables HNF to convert Non-cacheable requests to cacheable if MTE tags are required	RW	1'b1
15:8	Reserved	Reserved	RO	-
7	hnf_slc_repl_hint_use_en	1'b0: Interconnect generated SLC Replacement hints are used for eLRU. 1'b1: RN-F provided SLC Replacement hints are used for eLRU.	RW	1'b1
6:5	hnf_mpam_softlim_value	Soft Limit value for MPAM capacity partitioning. 2'b00: Soft limit is 0% below hardlimit. 2'b01: Soft limit is 3.13% (1/32) below hardlimit 2'b10: Soft limit is 6.25% (1/16) below hardlimit 2'b11: Soft limit is 9.38% (3/32) below hardlimit NOTE: Default is 3.13% below hardlimit. If CMAX value set is at or below 12.5%, soft limit is ignored.	RW	2'b01
4	hnf_force_flush_pcmo_deep_en	Make PCMO request for SLC and SF flush generated SN writes as Deep PCMO. CONSTRAINT: hnf_force_flush_pcmo_deep_en is valid only if hnf_force_flush_pcmo_en bit is set. CONSTRAINT: This bit can be set only if ALL SNs in the system support deep attribute.	RW	1'b0

Table 3-1216 por_hnf_por_hnf_r2_aux_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
3	hnf_force_flush_pcmo_en	Generate PCMO request for SLC and SF flush generated SN writes	RW	1'b0
2	hnf_force_flush_ewa0_en	Force SLC and SF flush to use EWA 0 for SN writes	RW	1'b0
1	Reserved	Reserved	RO	-
0	hnf_chic_rdnosnpsep_dis	Disables separation of Data and Comp in CHIC mode	RW	1'b0

por_hnf_cbusy_limit_ctl

Cbusy threshold limits for POCQ entries.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA18
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

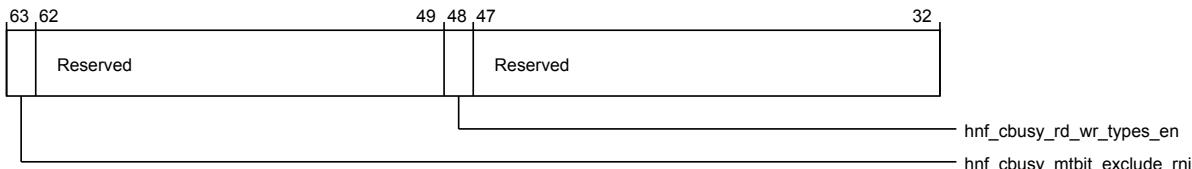


Figure 3-1197 por_hnf_por_hnf_cbusy_limit_ctl (high)

The following table shows the por_hnf_cbusy_limit_ctl higher register bit assignments.

Table 3-1217 por_hnf_por_hnf_cbusy_limit_ctl (high)

Bits	Field name	Description	Type	Reset
63	hnf_cbusy_mtbit_exclude_rni	Exclude RNI sources in multi-source mode	RW	1'b0
62:49	Reserved	Reserved	RO	-
48	hnf_cbusy_rd_wr_types_en	When set, CBusy for Reads and Writes are handled independently. The thresholds specified in this register are used for Read request types in POCQ	RW	1'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

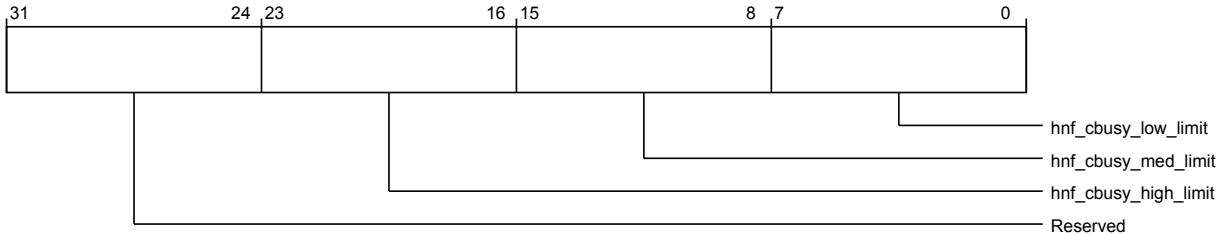


Figure 3-1198 por_hnf_por_hnf_cbusy_limit_ctl (low)

The following table shows the por_hnf_cbusy_limit_ctl lower register bit assignments.

Table 3-1218 por_hnf_por_hnf_cbusy_limit_ctl (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_cbusy_high_limit	POCQ limit for CBusy High	RW	Configuration dependent
15:8	hnf_cbusy_med_limit	POCQ limit for CBusy Med	RW	Configuration dependent
7:0	hnf_cbusy_low_limit	POCQ limit for CBusy Low	RW	Configuration dependent

por_hnf_ppu_pwpr

Functions as the power policy register for HN-F.

Its characteristics are:

Type RW

Register width (Bits) 32

Address offset 16'h1C00

Register reset 32'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.ppu

The following image shows the lower register bit assignments.

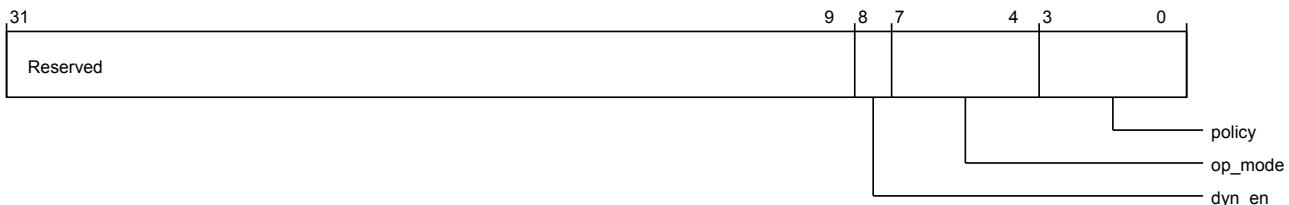


Figure 3-1199 por_hnf_ppu_pwpr

The following table shows the por_hnf_ppu_pwpr register bit assignments.

Table 3-1219 por_hnf_por_hnf_ppu_pwpr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	dyn_en	Dynamic transition enable	RW	1'b0
7:4	op_mode	HN-F operational power mode 4'b0011: FAM 4'b0010: HAM 4'b0001: SFONLY 4'b0000: NOSFSLC	RW	4'b0
3:0	policy	HN-F power mode policy 4'b1000: ON 4'b0111: FUNC_RET 4'b0010: MEM_RET 4'b0000: OFF	RW	4'b0

por_hnf_ppu_pwsr

Provides power status information for HN-F.

Its characteristics are:

Type RO

Register width (Bits) 32

Address offset 16'h10

Register reset 32'b0

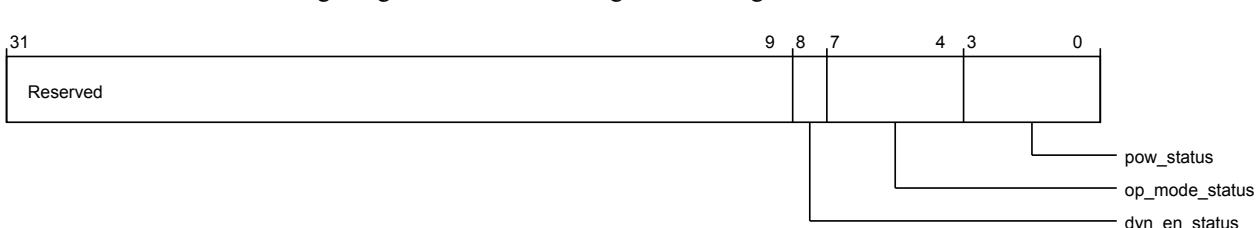


Figure 3-1200 por hnf por hnf ppu pwsr

The following table shows the por hnf ppu pwsr register bit assignments.

Table 3-1220 por hnf por hnf ppu pwsr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	dyn_en_status	Dynamic transition status	RO	1'b0

Table 3-1220 por_hnf_por_hnf_ppu_pwsr (low) (continued)

Bits	Field name	Description	Type	Reset
7:4	op_mode_status	HN-F operational mode status 4'b0011: FAM 4'b0010: HAM 4'b0001: SFONLY 4'b0000: NOSFSLC	RO	4'b0
3:0	pow_status	HN-F power mode status 4'b1000: ON 4'b0111: FUNC_RET 4'b0010: MEM_RET 4'b0000: OFF	RO	4'b0

por_hnf_ppu_misr

Functions as the power miscellaneous input current status register for HN-F.

Its characteristics are:

Type RO

Register width (Bits) 32

Address offset 16'h1C14

Register reset 32'b0

Usage constraints There are no usage constraints.

The following image shows the lower register bit assignments.

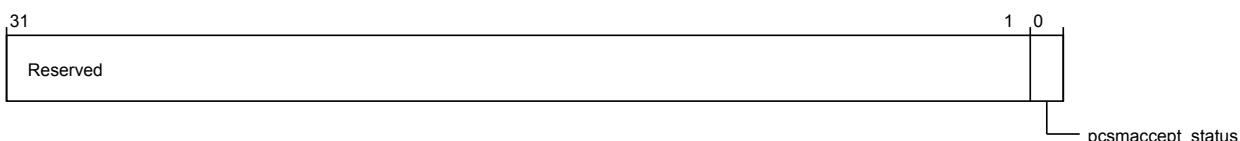


Figure 3-1201 por_hnf_por_hnf_ppu_misr

The following table shows the por_hnf_ppu_misr register bit assignments.

Table 3-1221 por_hnf_por_hnf_ppu_misr (low)

Bits	Field name	Description	Type	Reset
31:1	Reserved	Reserved	RO	-
0	pcsmaccept_status	HN-F RAM PCSMACCEPT status	RO	1'b0

por_hnf_ppu_idr0

Provides identification information for the HN-F PPU.

Its characteristics are:

Type	RO
Register width (Bits)	32
Address offset	16'h2BB0
Register reset	32'b00100000000011010010101000
Usage constraints	There are no usage constraints.

The following image shows the lower register bit assignments.

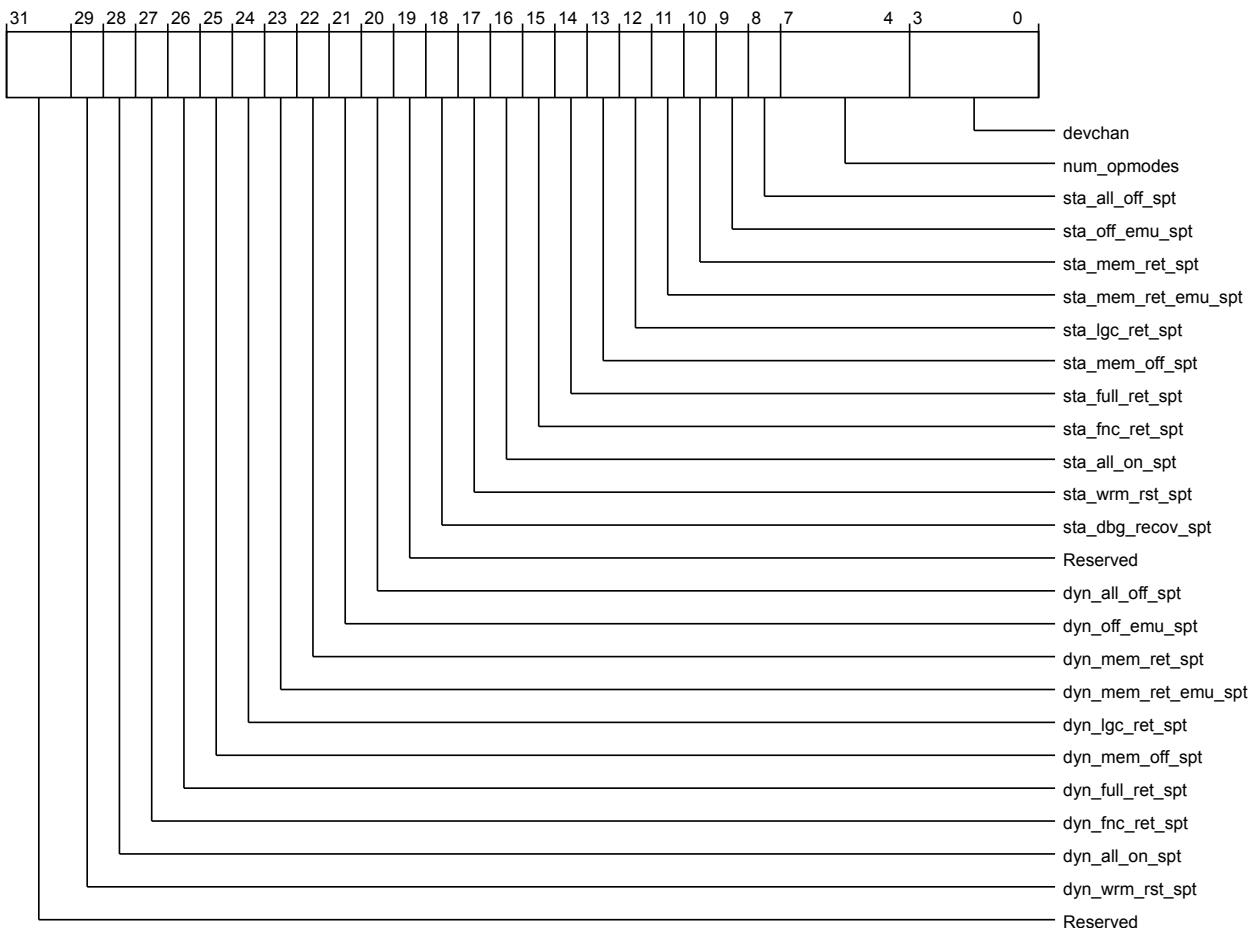


Figure 3-1202 por_hnf_por_hnf_ppu_idr0

The following table shows the por_hnf_ppu_idr0 register bit assignments.

Table 3-1222 por_hnf_por_hnf_ppu_idr0 (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29	dyn_wrm_RST_spt	Dynamic warm_RST support	RO	1'b0
28	dyn_all_on_spt	Dynamic on support	RO	1'b0
27	dyn_fnc_ret_spt	Dynamic func_ret support	RO	1'b1
26	dyn_full_ret_spt	Dynamic full_ret support	RO	1'b0

Table 3-1222 por_hnf_por_hnf_ppu_idr0 (low) (continued)

Bits	Field name	Description	Type	Reset
25	dyn_mem_off_spt	Dynamic mem_off support	RO	1'b0
24	dyn_lgc_ret_spt	Dynamic logic_ret support	RO	1'b0
23	dyn_mem_ret_emu_spt	Dynamic mem_ret_emu support	RO	1'b0
22	dyn_mem_ret_spt	Dynamic mem_ret support	RO	1'b0
21	dyn_off_emu_spt	Dynamic off_emu support	RO	1'b0
20	dyn_all_off_spt	Dynamic off support	RO	1'b0
19	Reserved	Reserved	RO	-
18	sta_dbg_recov_spt	Static dbg_recov support	RO	1'b0
17	sta_wrm_RST_spt	Static warm_RST support	RO	1'b0
16	sta_all_on_spt	Static on support	RO	1'b1
15	sta_fnc_ret_spt	Static func_ret support	RO	1'b1
14	sta_full_ret_spt	Static full_ret support	RO	1'b0
13	sta_mem_off_spt	Static mem_off support	RO	1'b1
12	sta_lgc_ret_spt	Static logic_ret support	RO	1'b0
11	sta_mem_ret_emu_spt	Static mem_ret_emu support	RO	1'b0
10	sta_mem_ret_spt	Static mem_ret support	RO	1'b1
9	sta_off_emu_spt	Static off_emu support	RO	1'b0
8	sta_all_off_spt	Static off support	RO	1'b1
7:4	num_opmodes	Number of operational modes	RO	4'b0100
3:0	devchan	Number of device interface channels	RO	1'b0

por_hnf_ppu_idr1

Provides identification information for the HN-F PPU.

Its characteristics are:

Type	RO
Register width (Bits)	32
Address offset	16'h2BB4
Register reset	32'b0
Usage constraints	There are no usage constraints.

The following image shows the lower register bit assignments.

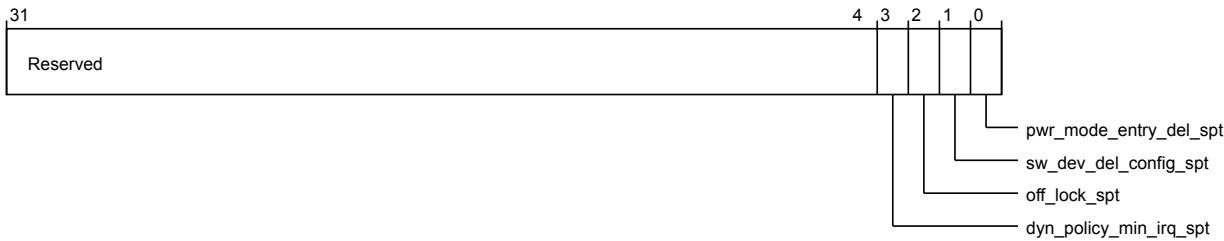


Figure 3-1203 por_hnf_por_hnf_ppu_idr1

The following table shows the por_hnf_ppu_idr1 register bit assignments.

Table 3-1223 por_hnf_por_hnf_ppu_idr1 (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	dyn_policy_min_irq_spt	Dynamic minimum policy interrupt support	RO	1'b0
2	off_lock_spt	Off and mem_ret lock support	RO	1'b0
1	sw_dev_del_config_spt	Software device delay control configuration support	RO	1'b0
0	pwr_mode_entry_del_spt	Power mode entry delay support	RO	1'b0

por_hnf_ppu_iidr

Functions as the power implementation identification register for HN-F.

Its characteristics are:

Type	RO
Register width (Bits)	32
Address offset	16'h2BC8
Register reset	32'b00001001110000000000000100111011
Usage constraints	There are no usage constraints.

The following image shows the lower register bit assignments.

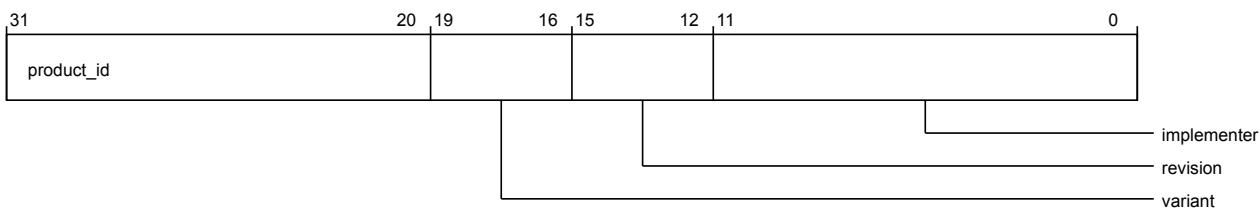


Figure 3-1204 por_hnf_por_hnf_ppu_iidr

The following table shows the por_hnf_ppu_iidr register bit assignments.

Table 3-1224 por_hnf_por_hnf_ppu_iidr (low)

Bits	Field name	Description	Type	Reset
31:20	product_id	Implementation identifier	RO	12'h434
19:16	variant	Implementation variant	RO	4'h0
15:12	revision	Implementation revision	RO	4'h0
11:0	implementer	Arm implementation	RO	12'h43B

por_hnf_ppu_aidr

Functions as the power architecture identification register for HN-F.

Its characteristics are:

Type RO

Register width (Bits) 32

Address offset 16'h2BCC

Register reset 32'b000010001

Usage constraints There are no usage constraints.

The following image shows the lower register bit assignments.

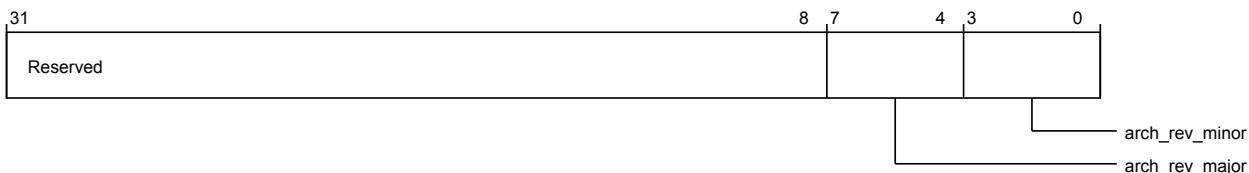


Figure 3-1205 por_hnf_por_hnf_ppu_aidr

The following table shows the por_hnf_ppu_aidr register bit assignments.

Table 3-1225 por_hnf_por_hnf_ppu_aidr (low)

Bits	Field name	Description	Type	Reset
31:8	Reserved	Reserved	RO	-
7:4	arch_rev_major	PPU architecture major revision	RO	4'h1
3:0	arch_rev_minor	PPU architecture minor revision	RO	4'h1

por_hnf_ppu_dyn_ret_threshold

Configures the dynamic retention threshold for SLC and SF RAM.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1D00

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.ppu

The following image shows the higher register bit assignments.



Figure 3-1206 por_hnf_por_hnf_ppu_dyn_ret_threshold (high)

The following table shows the por_hnf_ppu_dyn_ret_threshold higher register bit assignments.

Table 3-1226 por_hnf_por_hnf_ppu_dyn_ret_threshold (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

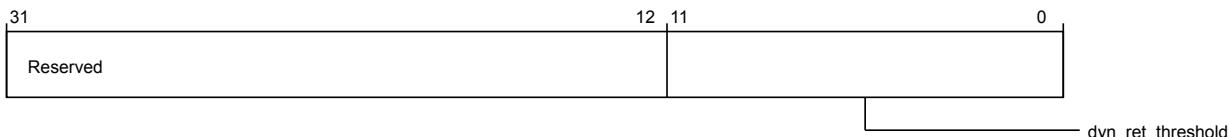


Figure 3-1207 por_hnf_por_hnf_ppu_dyn_ret_threshold (low)

The following table shows the por_hnf_ppu_dyn_ret_threshold lower register bit assignments.

Table 3-1227 por_hnf_por_hnf_ppu_dyn_ret_threshold (low)

Bits	Field name	Description	Type	Reset
31:12	Reserved	Reserved	RO	-
11:0	dyn_ret_threshold	HN-F RAM idle cycle count threshold	RW	32'b0

por_hnf_qos_band

Provides QoS classifications based on the QoS value ranges.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'hA80
Register reset	64'b1111111111011001011100001110000
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-1208 por_hnf_por_hnf_qos_band (high)

The following table shows the por_hnf_qos_band higher register bit assignments.

Table 3-1228 por_hnf_por_hnf_qos_band (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

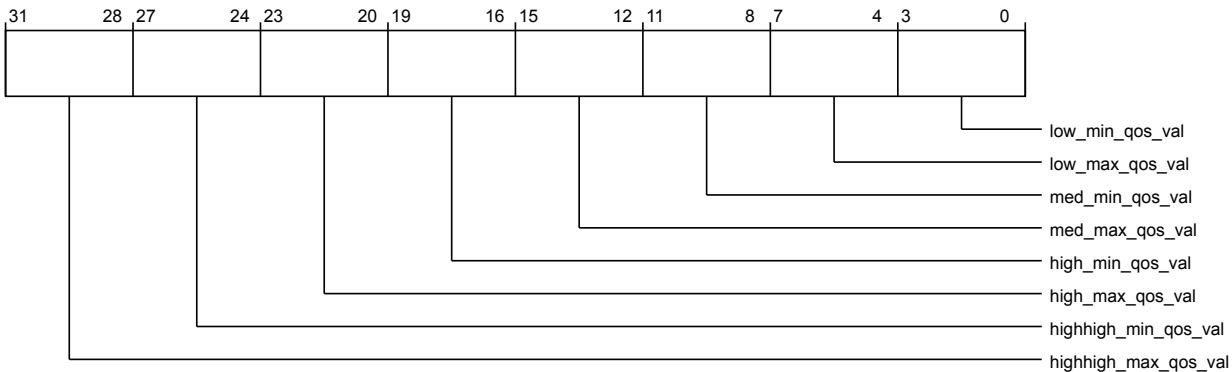


Figure 3-1209 por_hnf_por_hnf_qos_band (low)

The following table shows the por_hnf_qos_band lower register bit assignments.

Table 3-1229 por_hnf_por_hnf_qos_band (low)

Bits	Field name	Description	Type	Reset
31:28	highhigh_max_qos_val	Maximum value for HighHigh QoS class	RO	4'hF
27:24	highhigh_min_qos_val	Minimum value for HighHigh QoS class	RO	4'hF
23:20	high_max_qos_val	Maximum value for High QoS class	RO	4'hE
19:16	high_min_qos_val	Minimum value for High QoS class	RO	4'hC
15:12	med_max_qos_val	Maximum value for Medium QoS class	RO	4'hB
11:8	med_min_qos_val	Minimum value for Medium QoS class	RO	4'h8
7:4	low_max_qos_val	Maximum value for Low QoS class	RO	4'h7
3:0	low_min_qos_val	Minimum value for Low QoS class	RO	4'h0

por_hnf_qos_reservation

Controls POCQ maximum occupancy counts for SEQ class.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hA88
Register reset	64'b0000001
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

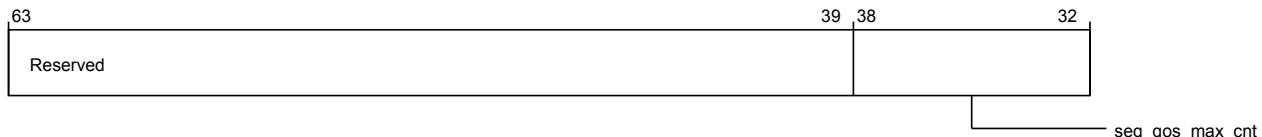


Figure 3-1210 por_hnf_por_hnf_qos_reservation (high)

The following table shows the por_hnf_qos_reservation higher register bit assignments.

Table 3-1230 por_hnf_por_hnf_qos_reservation (high)

Bits	Field name	Description	Type	Reset
63:39	Reserved	Reserved	RO	-
38:32	seq_qos_max_cnt	Number of entries reserved for SF evictions in POCQ. CONSTRAINT: Sum of dedicated entries for classes and SEQ can not exceed HNF_NUM_ENTRIES_POCQ_PARAM CONSTRAINT: Maximum number is 2 entries.	RW	6'h1

The following image shows the lower register bit assignments.

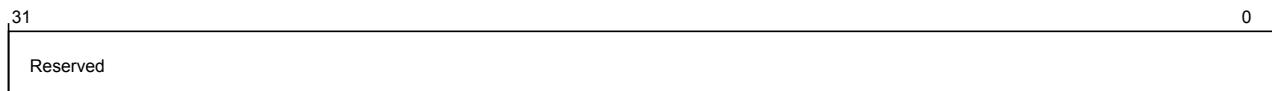


Figure 3-1211 por_hnf_por_hnf_qos_reservation (low)

The following table shows the por_hnf_qos_reservation lower register bit assignments.

Table 3-1231 por_hnf_por_hnf_qos_reservation (low)

Bits	Field name	Description	Type	Reset
31:0	Reserved	Reserved	RO	-

por_hnf_errfr

Functions as the error feature register.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'h3000
Register reset	64'b1001010100101
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.



Figure 3-1212 por_hnf_por_hnf_errfr (high)

The following table shows the por_hnf_errfr higher register bit assignments.

Table 3-1232 por_hnf_por_hnf_errfr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

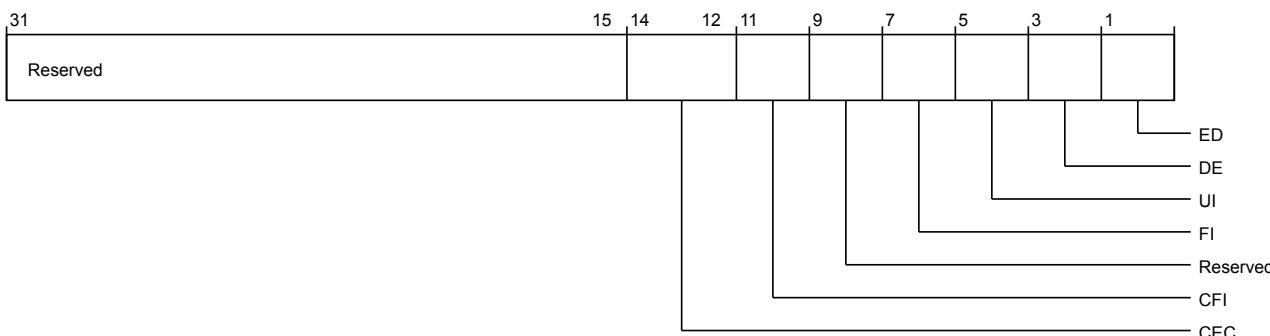


Figure 3-1213 por_hnf_por_hnf_errfr (low)

The following table shows the por_hnf_errfr lower register bit assignments.

Table 3-1233 por_hnf_por_hnf_errfr (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_hnf_errmisc[39:32] 3'b100: Implements 16-bit error counter in por_hnf_errmisc[47:32]	RO	3'b100

Table 3-1233 por_hnf_por_hnf_errfr (low) (continued)

Bits	Field name	Description	Type	Reset
11:10	CFI	Corrected error interrupt	RO	2'b10
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_hnf_errctlr

Functions as the error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3008

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

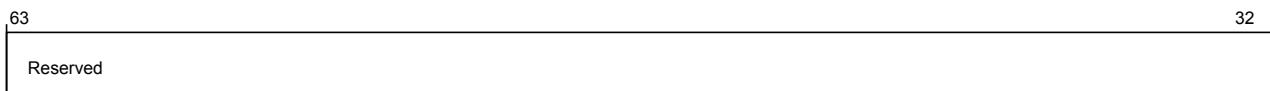


Figure 3-1214 por_hnf_por_hnf_errctlr (high)

The following table shows the por_hnf_errctlr higher register bit assignments.

Table 3-1234 por_hnf_por_hnf_errctlr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

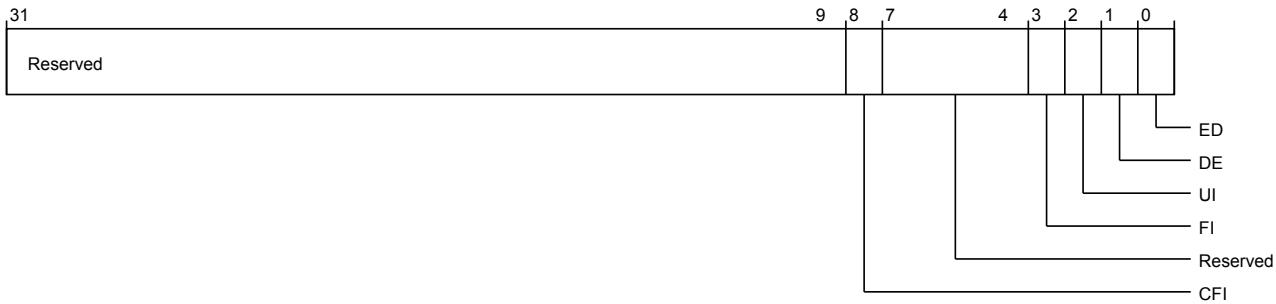


Figure 3-1215 por_hnf_por_hnf_errctlr (low)

The following table shows the por_hnf_errctlr lower register bit assignments.

Table 3-1235 por_hnf_por_hnf_errctlr (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_hnf_errfr.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_hnf_errfr.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_hnf_errfr.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_hnf_errfr.DE	RW	1'b0
0	ED	Enables error detection as specified in por_hnf_errfr.ED	RW	1'b0

por_hnf_errstatus

Functions as the error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type	W1C
Register width (Bits)	64
Address offset	16'h3010
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

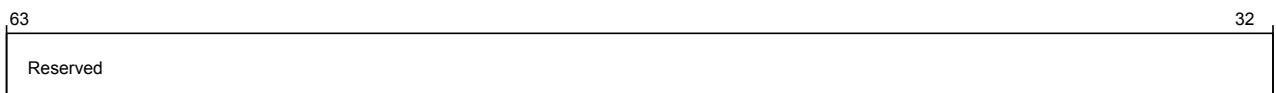


Figure 3-1216 por_hnf_por_hnf_errstatus (high)

The following table shows the por_hnf_errstatus higher register bit assignments.

Table 3-1236 por_hnf_por_hnf_errstatus (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-1217 por_hnf_por_hnf_errstatus (low)

The following table shows the por_hnf_errstatus lower register bit assignments.

Table 3-1237 por_hnf_por_hnf_errstatus (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_hnf_erraddr contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0

Table 3-1237 por_hnf_por_hnf_errstatus (low) (continued)

Bits	Field name	Description	Type	Reset
26	MV	por_hnf_errmisc valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_hnf_erraddr

Contains the error record address.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3018

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group por_hnf_secure_register_groups_override.ras_secure_access_override
override

The following image shows the higher register bit assignments.

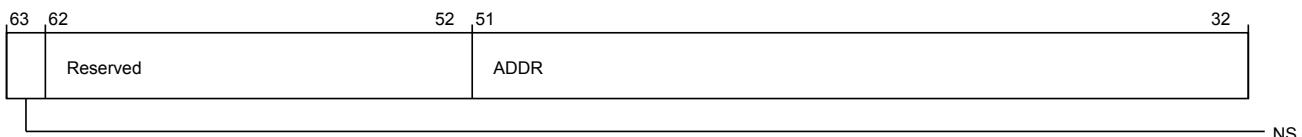


Figure 3-1218 por_hnf_por_hnf_erraddr (high)

The following table shows the por_hnf_erraddr higher register bit assignments.

Table 3-1238 por_hnf_por_hnf_erraddr (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_hnf_erraddr.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.

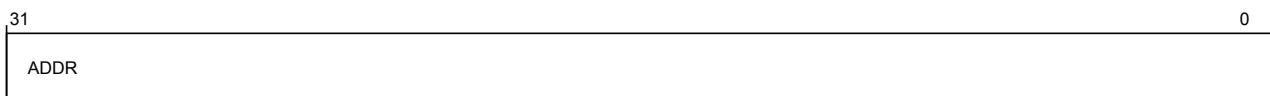


Figure 3-1219 por_hnf_por_hnf_erraddr (low)

The following table shows the por_hnf_erraddr lower register bit assignments.

Table 3-1239 por_hnf_por_hnf_erraddr (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_hnf_errmisc

Functions as the miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3020
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.ras_secure_access_override

The following image shows the higher register bit assignments.

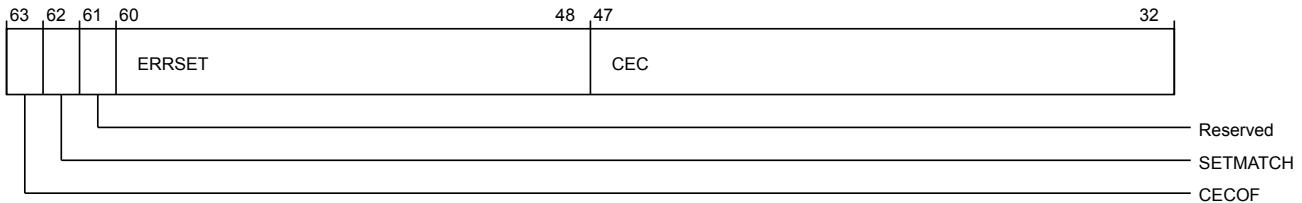


Figure 3-1220 por_hnf_por_hnf_errmisc (high)

The following table shows the por_hnf_errmisc higher register bit assignments.

Table 3-1240 por_hnf_por_hnf_errmisc (high)

Bits	Field name	Description	Type	Reset
63	CECOF	Corrected error counter overflow	RW	1'b0
62	SETMATCH	Set address match	RW	1'b0
61	Reserved	Reserved	RO	-
60:48	ERRSET	SLC/SF set address for ECC error	RW	13'b0
47:32	CEC	Corrected ECC error count	RW	16'b0

The following image shows the lower register bit assignments.

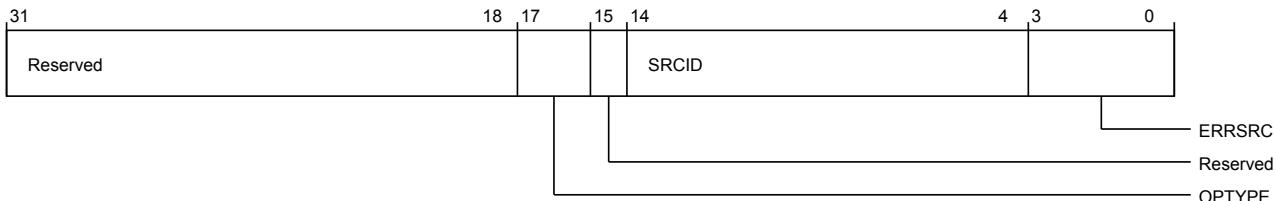


Figure 3-1221 por_hnf_por_hnf_errmisc (low)

The following table shows the por_hnf_errmisc lower register bit assignments.

Table 3-1241 por_hnf_por_hnf_errmisc (low)

Bits	Field name	Description	Type	Reset
31:18	Reserved	Reserved	RO	-
17:16	OPTYPE	Error op type 2'b00: Writes, CleanShared, Atomics and stash requests with invalid targets 2'b01: WriteBack, Evict, and Stash requests with valid target 2'b10: CMO 2'b11: Other op types	RW	2'b00

Table 3-1241 por_hnf_por_hnf_errmisc (low) (continued)

Bits	Field name	Description	Type	Reset
15	Reserved	Reserved	RO	-
14:4	SRCID	Error source ID	RW	11'b0
3:0	ERRSRC	Error source 4'b0001: Data single-bit ECC 4'b0010: Data double-bit ECC 4'b0011: Single-bit ECC overflow 4'b0100: Tag single-bit ECC 4'b0101: Tag double-bit ECC 4'b0111: SF tag single-bit ECC 4'b1000: SF tag double-bit ECC 4'b1010: Data parity error 4'b1011: Data parity and poison 4'b1100: NDE	RW	4'b0000

por_hnf_err_inj

Enables error injection and setup. When enabled for a given source ID and logic processor ID, HN-F returns a slave error and reports an error interrupt. This error interrupt emulates a SLC double-bit data ECC error. This feature enables software to test the error handler. The slave error is reported for cacheable read access for which SLC hit is the data source. No slave error or error interrupt is reported for cacheable read access in which SLC miss is the data source.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3030
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

The following image shows the higher register bit assignments.



Figure 3-1222 por_hnf_por_hnf_err_inj (high)

The following table shows the por_hnf_err_inj higher register bit assignments.

Table 3-1242 por_hnf_por_hnf_err_inj (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

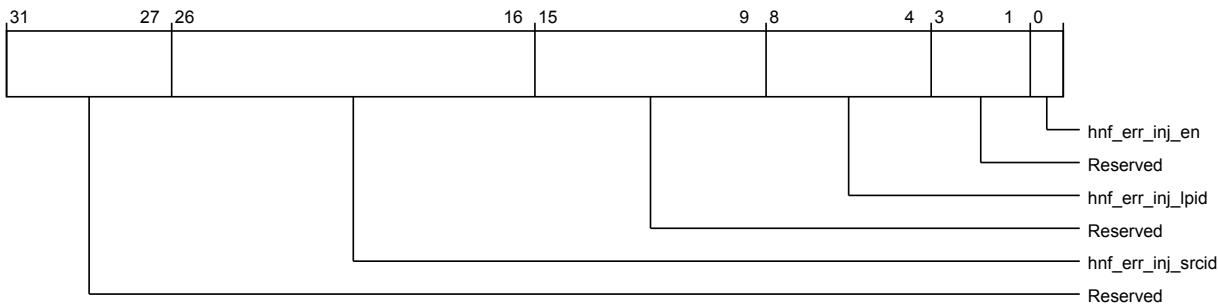


Figure 3-1223 por_hnf_por_hnf_err_inj (low)

The following table shows the por_hnf_err_inj lower register bit assignments.

Table 3-1243 por_hnf_por_hnf_err_inj (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:16	hnf_err_inj_srcid	RN source ID for read access which results in a SLC miss; does not report slave error or error to match error injection	RW	11'h0
15:9	Reserved	Reserved	RO	-
8:4	hnf_err_inj_lpid	LPID used to match for error injection	RW	5'h0
3:1	Reserved	Reserved	RO	-
0	hnf_err_inj_en	Enables error injection and report	RW	1'b0

por_hnf_byte_par_err_inj

Functions as the byte parity error injection register for HN-F.

Its characteristics are:

Type WO

Register width (Bits) 64

Address offset 16'h3038

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

The following image shows the higher register bit assignments.

63

Reserved

32

Figure 3-1224 por_hnf_por_hnf_byte_par_err_inj (high)

The following table shows the por_hnf_byte_par_err_inj higher register bit assignments.

Table 3-1244 por_hnf_por_hnf_byte_par_err_inj (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31

Reserved

5

4

0

Figure 3-1225 por_hnf_por_hnf_byte_par_err_inj (low)

The following table shows the por_hnf_byte_par_err_inj lower register bit assignments.

Table 3-1245 por_hnf_por_hnf_byte_par_err_inj (low)

Bits	Field name	Description	Type	Reset
31:5	Reserved	Reserved	RO	-
4:0	hnf_byte_par_err_inj	Specifies a byte lane; once this register is written, a byte parity error is injected in the specified byte lane on the next SLC hit; the error will be injected in all data flits on specified byte (0 to 31)	WO	5'h0

por_hnf_errfr_NS

Functions as the non-secure error feature register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'h3100

Register reset 64'b1001010100101

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

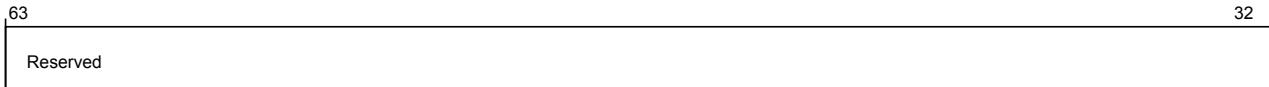


Figure 3-1226 por_hnf_por_hnf_errfr_ns (high)

The following table shows the por_hnf_errfr_NS higher register bit assignments.

Table 3-1246 por_hnf_por_hnf_errfr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

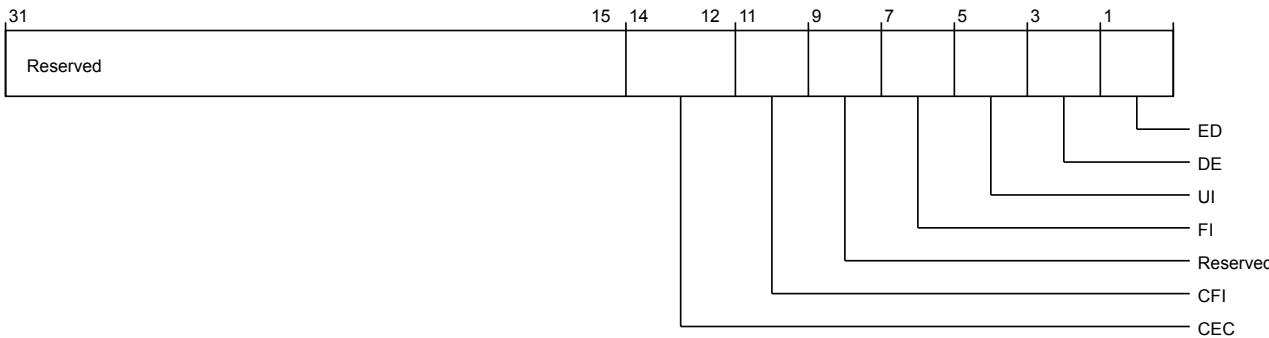


Figure 3-1227 por_hnf_por_hnf_errfr_ns (low)

The following table shows the por_hnf_errfr_NS lower register bit assignments.

Table 3-1247 por_hnf_por_hnf_errfr_ns (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:12	CEC	Standard corrected error count mechanism 3'b000: Does not implement standardized error counter model 3'b010: Implements 8-bit error counter in por_hnf_errmisc_NS[39:32] 3'b100: Implements 16-bit error counter in por_hnf_errmisc_NS[47:32]	RO	3'b100
11:10	CFI	Corrected error interrupt	RO	2'b10
9:8	Reserved	Reserved	RO	-
7:6	FI	Fault handling interrupt	RO	2'b10
5:4	UI	Uncorrected error interrupt	RO	2'b10
3:2	DE	Deferred errors for data poison	RO	2'b01
1:0	ED	Error detection	RO	2'b01

por_hnf_errctlr_NS

Functions as the non-secure error control register. Controls whether specific error-handling interrupts and error detection/deferment are enabled.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3108
Register reset	64'b0
Usage constraints	There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1228 por_hnf_por_hnf_errctlr_ns (high)

The following table shows the por_hnf_errctlr_NS higher register bit assignments.

Table 3-1248 por_hnf_por_hnf_errctlr_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

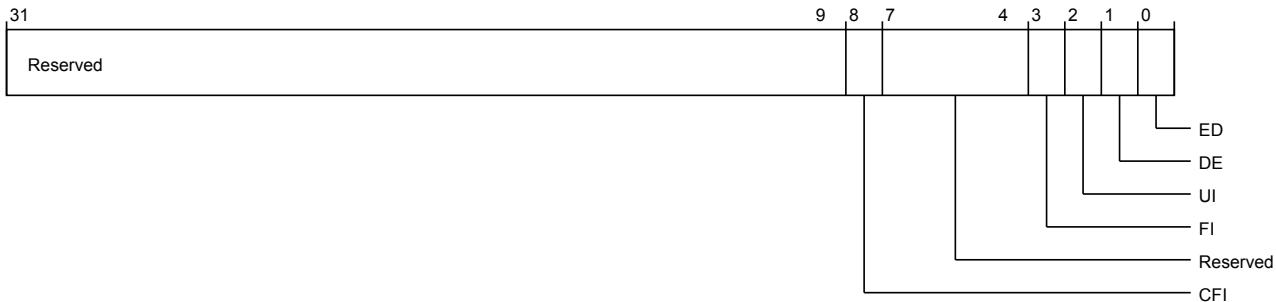


Figure 3-1229 por_hnf_por_hnf_errctlr_ns (low)

The following table shows the por_hnf_errctlr_NS lower register bit assignments.

Table 3-1249 por_hnf_por_hnf_errctlr_ns (low)

Bits	Field name	Description	Type	Reset
31:9	Reserved	Reserved	RO	-
8	CFI	Enables corrected error interrupt as specified in por_hnf_errfr_NS.CFI	RW	1'b0
7:4	Reserved	Reserved	RO	-

Table 3-1249 por_hnf_por_hnf_errctlr_ns (low) (continued)

Bits	Field name	Description	Type	Reset
3	FI	Enables fault handling interrupt for all detected deferred errors as specified in por_hnf_errfr_NS.FI	RW	1'b0
2	UI	Enables uncorrected error interrupt as specified in por_hnf_errfr_NS.UI	RW	1'b0
1	DE	Enables error deferment as specified in por_hnf_errfr_NS.DE	RW	1'b0
0	ED	Enables error detection as specified in por_hnf_errfr_NS.ED	RW	1'b0

por_hnf_errstatus_NS

Functions as the non-secure error status register. AV and MV bits must be cleared in the same cycle, otherwise the error record does not have a consistent view.

Its characteristics are:

Type W1C

Register width (Bits) 64

Address offset 16'h3110

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1230 por_hnf_por_hnf_errstatus_ns (high)

The following table shows the por_hnf_errstatus_NS higher register bit assignments.

Table 3-1250 por_hnf_por_hnf_errstatus_ns (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.



Figure 3-1231 por_hnf_por_hnf_errstatus_ns (low)

The following table shows the por_hnf_errstatus_NS lower register bit assignments.

Table 3-1251 por_hnf_por_hnf_errstatus_ns (low)

Bits	Field name	Description	Type	Reset
31	AV	Address register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Address is valid; por_hnf_erraddr_NS contains a physical address for that recorded error 1'b0: Address is not valid	W1C	1'b0
30	V	Register valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and are not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error recorded; register is valid 1'b0: No errors recorded	W1C	1'b0
29	UE	Uncorrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error detected that is not corrected and is not deferred to a slave 1'b0: No uncorrected errors detected	W1C	1'b0
28	Reserved	Reserved	RO	-
27	OF	Overflow; asserted when multiple errors of the highest priority type are detected; write a 1 to clear 1'b1: More than one error detected 1'b0: Only one error of the highest priority type detected as described by UE/DE/CE fields	W1C	1'b0
26	MV	por_hnf_errmisc_NS valid; writes to this bit are ignored if any of the UE, DE, or CE bits are set to 1, and the highest priority are not cleared to 0 in the same write; write a 1 to clear 1'b1: Miscellaneous registers are valid 1'b0: Miscellaneous registers are not valid	W1C	1'b0
25	Reserved	Reserved	RO	-

Table 3-1251 por_hnf_por_hnf_errstatus_ns (low) (continued)

Bits	Field name	Description	Type	Reset
24	CE	Corrected errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one transient corrected error recorded 1'b0: No corrected errors recorded	W1C	1'b0
23	DE	Deferred errors; writes to this bit are ignored if the OF bit is set to 1, and is not cleared to 0 in the same write; write a 1 to clear 1'b1: At least one error is not corrected and is deferred 1'b0: No errors deferred	W1C	1'b0
22:0	Reserved	Reserved	RO	-

por_hnf_erraddr_NS

Contains the non-secure error record address.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3118

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

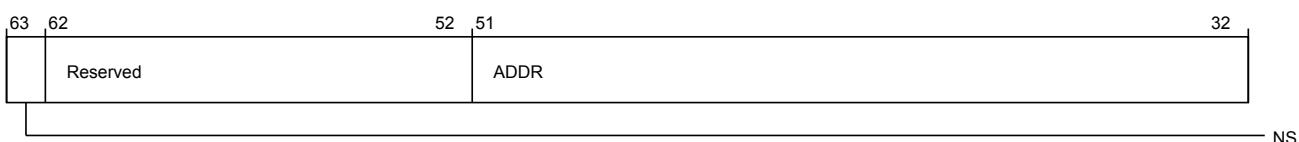


Figure 3-1232 por_hnf_por_hnf_erraddr_ns (high)

The following table shows the por_hnf_erraddr_NS higher register bit assignments.

Table 3-1252 por_hnf_por_hnf_erraddr_ns (high)

Bits	Field name	Description	Type	Reset
63	NS	Security status of transaction 1'b1: Non-secure transaction 1'b0: Secure transaction CONSTRAINT: por_hnf_erraddr_NS.NS is redundant. Since it is writable, it cannot be used for logic qualification.	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	ADDR	Transaction address	RW	52'b0

The following image shows the lower register bit assignments.

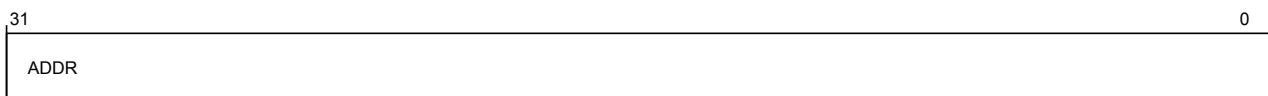


Figure 3-1233 por_hnf_por_hnf_erraddr_ns (low)

The following table shows the por_hnf_erraddr_NS lower register bit assignments.

Table 3-1253 por_hnf_por_hnf_erraddr_ns (low)

Bits	Field name	Description	Type	Reset
31:0	ADDR	Transaction address	RW	52'b0

por_hnf_errmisc_NS

Functions as the non-secure miscellaneous error register. Contains miscellaneous information about deferred/uncorrected errors.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3120

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

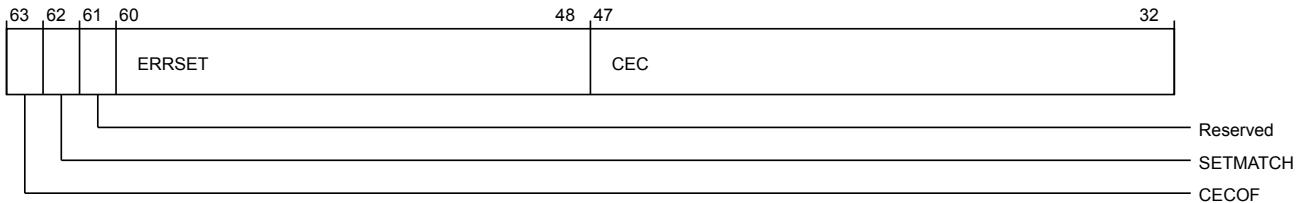


Figure 3-1234 por_hnf_errmisc_ns (high)

The following table shows the por_hnf_errmisc_NS higher register bit assignments.

Table 3-1254 por_hnf_errmisc_ns (high)

Bits	Field name	Description	Type	Reset
63	CECOF	Corrected error counter overflow	RW	1'b0
62	SETMATCH	Set address match	RW	1'b0
61	Reserved	Reserved	RO	-
60:48	ERRSET	SLC/SF set address for ECC error	RW	13'b0
47:32	CEC	Corrected ECC error count	RW	16'b0

The following image shows the lower register bit assignments.

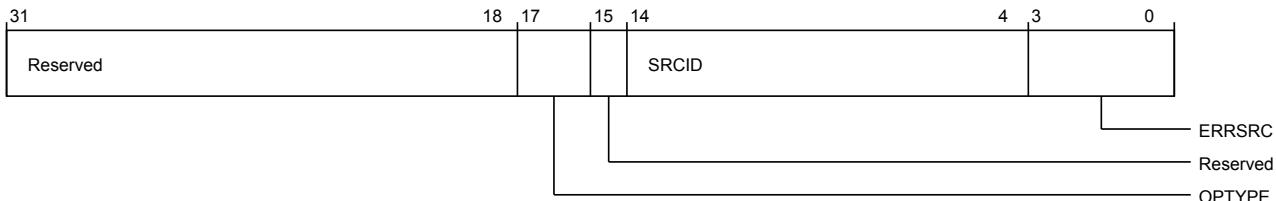


Figure 3-1235 por_hnf_errmisc_ns (low)

The following table shows the por_hnf_errmisc_NS lower register bit assignments.

Table 3-1255 por_hnf_errmisc_ns (low)

Bits	Field name	Description	Type	Reset
31:18	Reserved	Reserved	RO	-
17:16	OPTYPE	Error op type 2'b00: Writes, CleanShared, Atomics and stash requests with invalid targets 2'b01: WriteBack, Evict, and Stash requests with valid target 2'b10: CMO 2'b11: Other op types	RW	2'b00

Table 3-1255 por_hnf_por_hnf_errmisc_ns (low) (continued)

Bits	Field name	Description	Type	Reset
15	Reserved	Reserved	RO	-
14:4	SRCID	Error source ID	RW	11'b0
3:0	ERRSRC	Error source 4'b0001: Data single-bit ECC 4'b0010: Data double-bit ECC 4'b0011: Single-bit ECC overflow 4'b0100: Tag single-bit ECC 4'b0101: Tag double-bit ECC 4'b0111: SF tag single-bit ECC 4'b1000: SF tag double-bit ECC 4'b1010: Data parity error 4'b1011: Data parity and poison 4'b1100: NDE	RW	4'b0000

por_hnf_slc_lock_ways

Controls SLC way lock settings.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC00
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. The SLC must be flushed before writing to this register. Non-configuration accesses to this HN-F cannot occur before this configuration write and after the SLC flush.
Secure group override	por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.



Figure 3-1236 por_hnf_por_hnf_slc_lock_ways (high)

The following table shows the por_hnf_slc_lock_ways higher register bit assignments.

Table 3-1256 por_hnf_por_hnf_slc_lock_ways (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

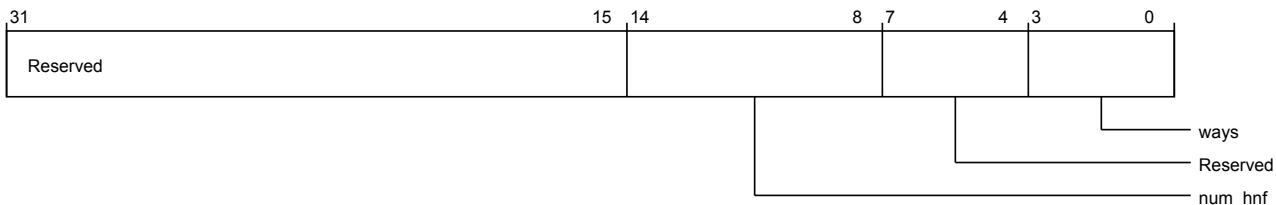


Figure 3-1237 por_hnf_por_hnf_slc_lock_ways (low)

The following table shows the por_hnf_slc_lock_ways lower register bit assignments.

Table 3-1257 por_hnf_por_hnf_slc_lock_ways (low)

Bits	Field name	Description	Type	Reset
31:15	Reserved	Reserved	RO	-
14:8	num_hnf	Number of HN-Fs in NUMA (non-uniform memory access) region	RW	Configuration dependent
7:4	Reserved	Reserved	RO	-
3:0	ways	Number of SLC ways locked (1, 2, 4, 8, 12)	RW	4'b0

por_hnf_slc_lock_base0

Functions as the base register for lock region 0 [47:0].

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC08
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. The SLC must be flushed before writing to this register. Non-configuration accesses to this HN-F cannot occur before this configuration write and after the SLC flush.
Secure group override	por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

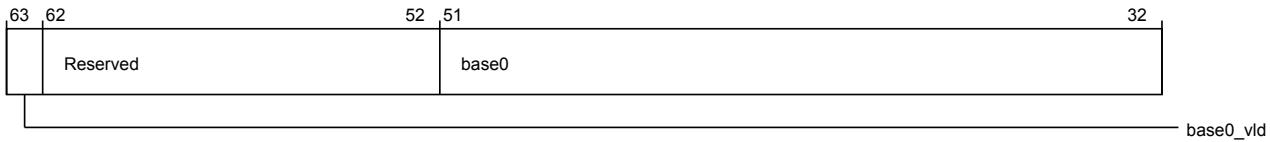


Figure 3-1238 por_hnf_por_hnf_slc_lock_base0 (high)

The following table shows the por_hnf_slc_lock_base0 higher register bit assignments.

Table 3-1258 por_hnf_por_hnf_slc_lock_base0 (high)

Bits	Field name	Description	Type	Reset
63	base0_vld	Lock region 0 base valid	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	base0	Lock region 0 base address	RW	52'b0

The following image shows the lower register bit assignments.

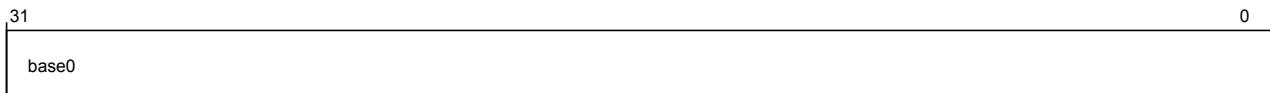


Figure 3-1239 por_hnf_por_hnf_slc_lock_base0 (low)

The following table shows the por_hnf_slc_lock_base0 lower register bit assignments.

Table 3-1259 por_hnf_por_hnf_slc_lock_base0 (low)

Bits	Field name	Description	Type	Reset
31:0	base0	Lock region 0 base address	RW	52'b0

por_hnf_slc_lock_base1

Functions as the base register for lock region 1 [47:0].

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC10

Register reset 64'b0

Usage constraints Only accessible by secure accesses. The SLC must be flushed before writing to this register. Non-configuration accesses to this HN-F cannot occur before this configuration write and after the SLC flush.

Secure group override por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

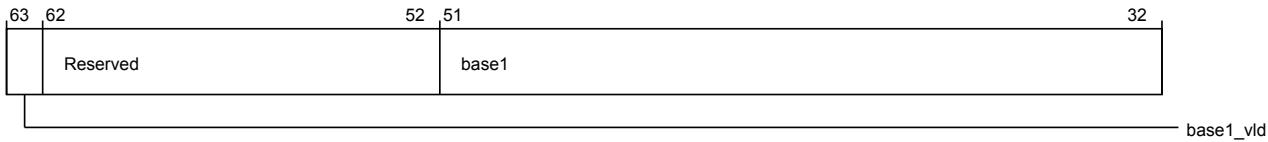


Figure 3-1240 por_hnf_por_hnf_slc_lock_base1 (high)

The following table shows the por_hnf_slc_lock_base1 higher register bit assignments.

Table 3-1260 por_hnf_por_hnf_slc_lock_base1 (high)

Bits	Field name	Description	Type	Reset
63	base1_vld	Lock region 1 base valid	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	base1	Lock region 1 base address	RW	52'b0

The following image shows the lower register bit assignments.

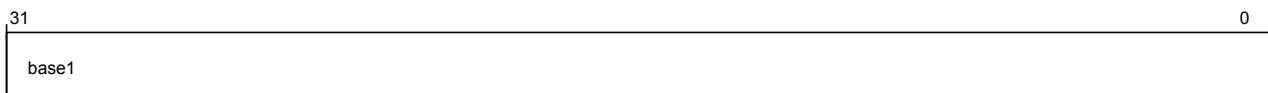


Figure 3-1241 por_hnf_por_hnf_slc_lock_base1 (low)

The following table shows the por_hnf_slc_lock_base1 lower register bit assignments.

Table 3-1261 por_hnf_por_hnf_slc_lock_base1 (low)

Bits	Field name	Description	Type	Reset
31:0	base1	Lock region 1 base address	RW	52'b0

por_hnf_slc_lock_base2

Functions as the base register for lock region 2 [47:0].

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC18

Register reset 64'b0

Usage constraints Only accessible by secure accesses. The SLC must be flushed before writing to this register. Non-configuration accesses to this HN-F cannot occur before this configuration write and after the SLC flush.

Secure group override por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

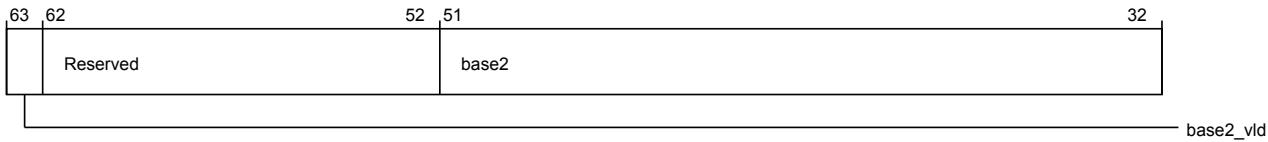


Figure 3-1242 por_hnf_por_hnf_slc_lock_base2 (high)

The following table shows the por_hnf_slc_lock_base2 higher register bit assignments.

Table 3-1262 por_hnf_por_hnf_slc_lock_base2 (high)

Bits	Field name	Description	Type	Reset
63	base2_vld	Lock region 2 base valid	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	base2	Lock region 2 base address	RW	52'b0

The following image shows the lower register bit assignments.

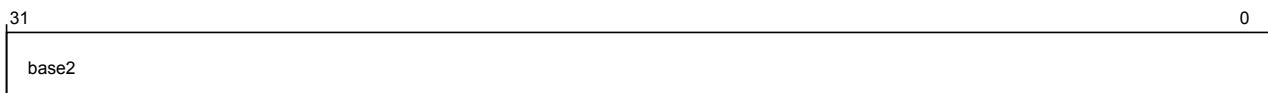


Figure 3-1243 por_hnf_por_hnf_slc_lock_base2 (low)

The following table shows the por_hnf_slc_lock_base2 lower register bit assignments.

Table 3-1263 por_hnf_por_hnf_slc_lock_base2 (low)

Bits	Field name	Description	Type	Reset
31:0	base2	Lock region 2 base address	RW	52'b0

por_hnf_slc_lock_base3

Functions as the base register for lock region 3 [47:0].

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC20

Register reset 64'b0

Usage constraints Only accessible by secure accesses. The SLC must be flushed before writing to this register. Non-configuration accesses to this HN-F cannot occur before this configuration write and after the SLC flush.

Secure group override por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

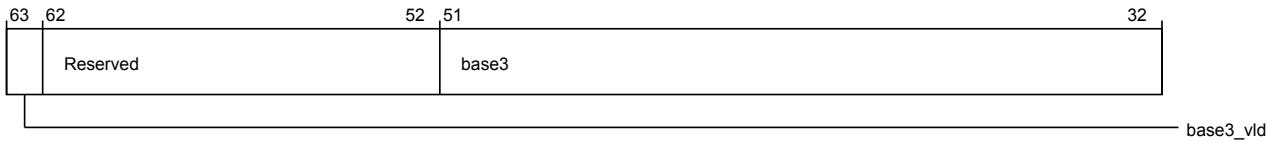


Figure 3-1244 por_hnf_por_hnf_slc_lock_base3 (high)

The following table shows the por_hnf_slc_lock_base3 higher register bit assignments.

Table 3-1264 por_hnf_por_hnf_slc_lock_base3 (high)

Bits	Field name	Description	Type	Reset
63	base3_vld	Lock region 3 base valid	RW	1'b0
62:52	Reserved	Reserved	RO	-
51:32	base3	Lock region 3 base address	RW	52'b0

The following image shows the lower register bit assignments.

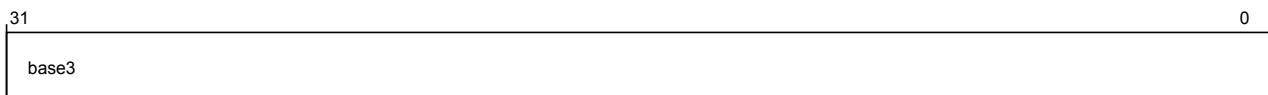


Figure 3-1245 por_hnf_por_hnf_slc_lock_base3 (low)

The following table shows the por_hnf_slc_lock_base3 lower register bit assignments.

Table 3-1265 por_hnf_por_hnf_slc_lock_base3 (low)

Bits	Field name	Description	Type	Reset
31:0	base3	Lock region 3 base address	RW	52'b0

por_hnf_rni_region_vec

Functions as the control register for RN-I source SLC way allocation.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC28
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

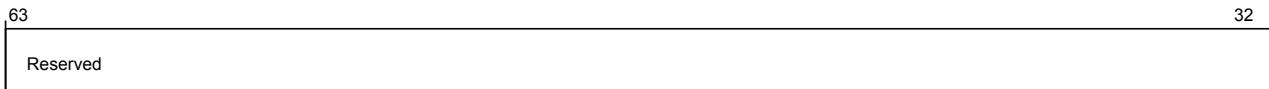


Figure 3-1246 por_hnf_por_hnf_rni_region_vec (high)

The following table shows the por_hnf_rni_region_vec higher register bit assignments.

Table 3-1266 por_hnf_por_hnf_rni_region_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

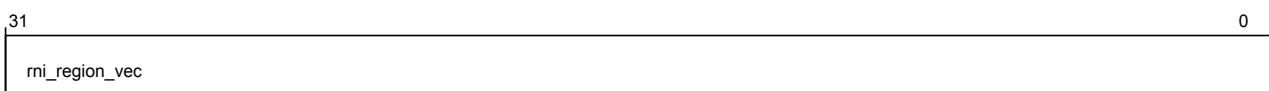


Figure 3-1247 por_hnf_por_hnf_rni_region_vec (low)

The following table shows the por_hnf_rni_region_vec lower register bit assignments.

Table 3-1267 por_hnf_por_hnf_rni_region_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rni_region_vec	Bit vector mask; identifies which logical IDs of the RN-Is to allocate to the locked region NOTE: Must be set to 32'b0 if range-based region locking or OCM is enabled.	RW	32'b0

por_hnf_rnd_region_vec

Functions as the control register for RN-D source SLC way allocation.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC30
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

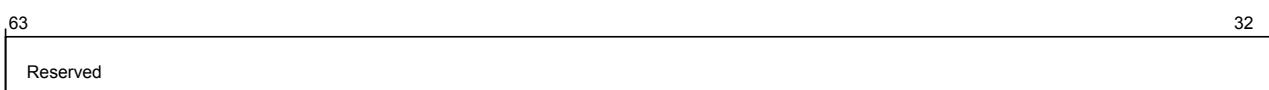


Figure 3-1248 por_hnf_por_hnf_rnd_region_vec (high)

The following table shows the por_hnf_rnd_region_vec higher register bit assignments.

Table 3-1268 por_hnf_por_hnf_rnd_region_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

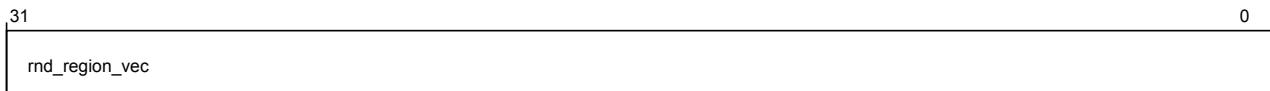


Figure 3-1249 por_hnf_por_hnf_rnd_region_vec (low)

The following table shows the por_hnf_rnd_region_vec lower register bit assignments.

Table 3-1269 por_hnf_por_hnf_rnd_region_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnd_region_vec	Bit vector mask; identifies which logical IDs of the RN-Ds to allocate to the locked region NOTE: Must be set to 32'b0 if range-based region locking or OCM is enabled.	RW	32'b0

por_hnf_rnf_region_vec

Functions as the control register for RN-F source SLC way allocation.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC38

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

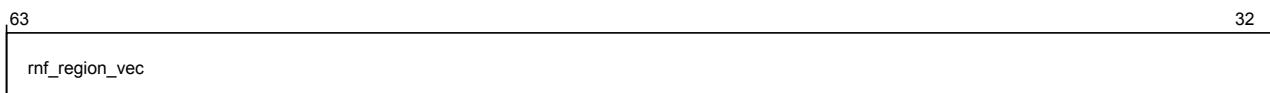


Figure 3-1250 por_hnf_por_hnf_rnf_region_vec (high)

The following table shows the por_hnf_rnf_region_vec higher register bit assignments.

Table 3-1270 por_hnf_por_hnf_rnf_region_vec (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_region_vec	Bit vector mask; identifies which logical IDs of the RN-Fs to allocate to the locked region NOTE: Must be 64'b0 if range-based region locking or OCM is enabled.	RW	64'b0

The following image shows the lower register bit assignments.

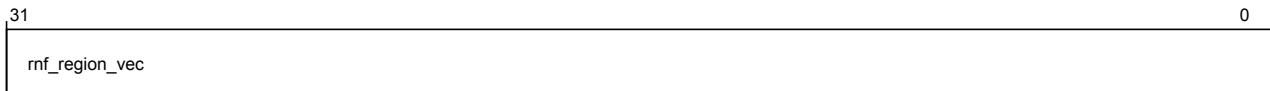


Figure 3-1251 por_hnf_por_hnf_rnf_region_vec (low)

The following table shows the por_hnf_rnf_region_vec lower register bit assignments.

Table 3-1271 por_hnf_por_hnf_rnf_region_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_region_vec	Bit vector mask; identifies which logical IDs of the RN-Fs to allocate to the locked region NOTE: Must be 64'b0 if range-based region locking or OCM is enabled.	RW	64'b0

por_hnf_rnf_region_vec1

Functions as the control register for RN-F source SLC way allocation for logical IDs 64 through 127.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC40

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.



Figure 3-1252 por_hnf_por_hnf_rnf_region_vec1 (high)

The following table shows the por_hnf_rnf_region_vec1 higher register bit assignments.

Table 3-1272 por_hnf_por_hnf_rnf_region_vec1 (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_region_vec1	Bit vector mask; identifies which logical IDs of the RN-Fs to allocate to the locked region NOTE: Must be 64'b0 if range-based region locking or OCM is enabled.	RW	64'b0

The following image shows the lower register bit assignments.

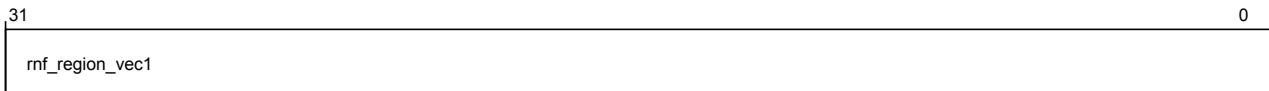


Figure 3-1253 por_hnf_por_hnf_rnf_region_vec1 (low)

The following table shows the por hnf rnf region vec1 lower register bit assignments.

Table 3-1273 por_hnf_por_hnf_rnf_region_vec1 (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_region_vec1	Bit vector mask; identifies which logical IDs of the RN-Fs to allocate to the locked region NOTE: Must be 64'b0 if range-based region locking or OCM is enabled.	RW	64'b0

por_hnf_slcway_partition0_rnf_vec

Functions as the control register for RN-Fs that can allocate to partition 0 (ways 0, 1, 2, and 3).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC48

Usage constraints	Only accessible by secure accesses.
Secure group	por_hnf_secure_register_groups_override.slc_lock_ways

override

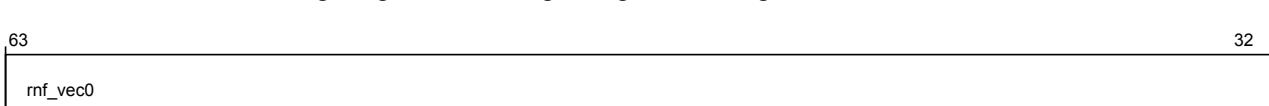


Figure 3-1254 por hnf por hnf slcway partition0 rnf vec (high)

The following table shows the por, hnf, slcway, partition0, rnf, vec higher register bit assignments

Table 3-1274 por hnf por hnf slcway partition0 rnf vec (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_vec0	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments.

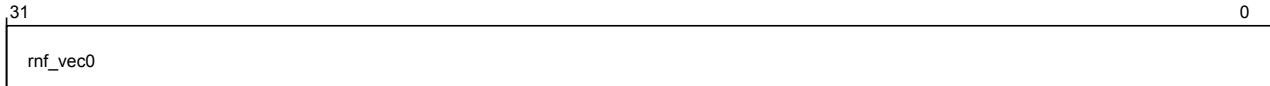


Figure 3-1255 por_hnf_por_hnf_slcway_partition0_rnf_vec (low)

The following table shows the port_hnf_slewway_partition0_rnf_vec lower register bit assignments.

Table 3-1275 por_hnf_por_hnf_slcway_partition0_rnf_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec0	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por_hnf_slcway_partition1_rnf_vec

Functions as the control register for RN-Fs that can allocate to partition 1 (ways 4, 5, 6, and 7).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC50

Usage constraints	Only accessible by secure accesses.
Secure group	por_hnf_secure_register_groups_override.slc_lock_ways

The following table summarizes the main findings.

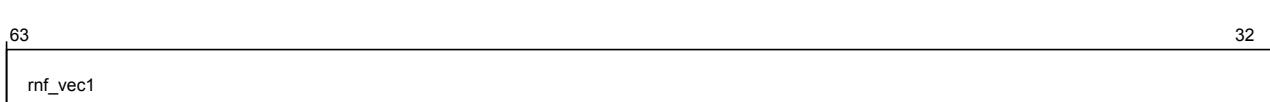


Figure 3-1256 por hnf por hnf slcway partition1 rnf vec (high)

The following table shows the port bit assignments for the `bfifoway_partition1_rnf_vec` higher register.

Table 3-1276 nor hnf nor hnf slcway partition1 rnf vec (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_vec1	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments

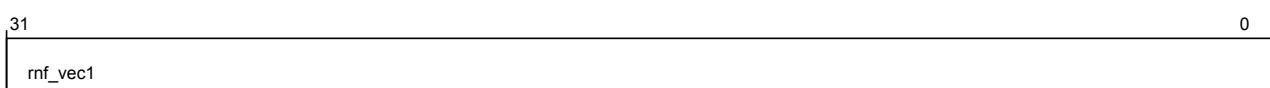


Figure 3-1257 por hnf por hnf slcway partition1 rnf vec (low)

The following table shows the por_hnf_slewway_partition1_rnf_vec lower register bit assignments.

Table 3-1277 por_hnf_por_hnf_slcway_partition1_rnf_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec1	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por_hnf_slcway_partition2_rnf_vec

Functions as the control register for RN-Fs that can allocate to partition 2 (ways 8, 9, 10, and 11).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC58

Usage constraints	Only accessible by secure accesses.
Secure group	por_hnf_secure_register_groups_override.slc_lock_ways

The following section describes the high-level architecture of the system.

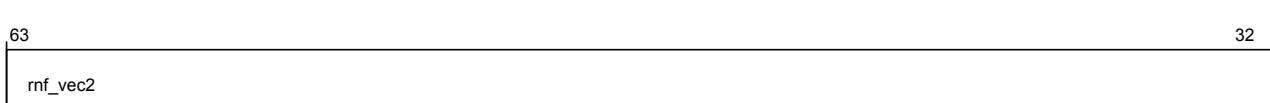


Figure 3-1258 por hnf por hnf slcway partition2 rnf vec (high)

The following table shows the por_hnf_slcway_partition2_rnf_vec higher register bit assignments

Table 3-1278 por hnf por hnf slcway partition2 rnf vec (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_vec2	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments

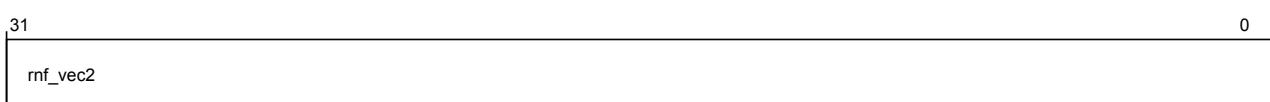


Figure 3-1259 por_hnf_por_hnf_slcway_partition2_rnf_vec (low)

The following table shows the por_hnf_slcway_partition2_rnf_vec_lower register bit assignments.

Table 3-1279 por_hnf_por_hnf_slcway_partition2_rnf_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec2	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por_hnf_slcway_partition3_rnf_vec

Functions as the control register for RN-Fs that can allocate to partition 3 (ways 12, 13, 14, and 15).

Its characteristics are:

The following image shows the higher register bit assignments.

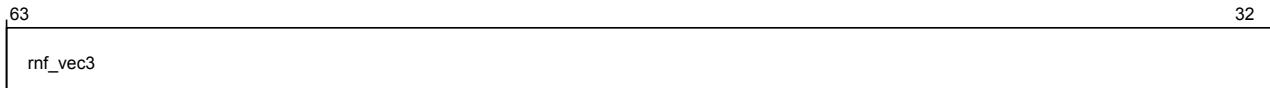


Figure 3-1260 por_hnf_por_hnf_slcway_partition3_rnf_vec (high)

The following table shows the port assignments for the higher register bit assignments.

Table 3-1280 por_hnf_por_hnf_slcway_partition3_rnf_vec (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_vec3	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments.

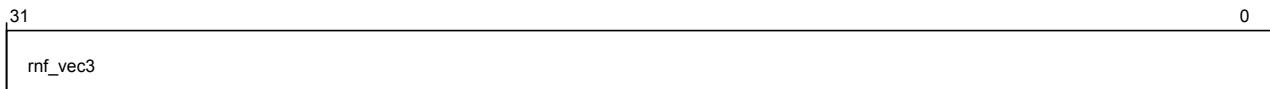


Figure 3-1261 por_hnf_por_hnf_slcway_partition3_rnf_vec (low)

The following table shows the port pin slew rate partition3 runt vector lower register bit assignments.

Table 3-1281 por_hnf_por_hnf_slcway_partition3_rnf_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec3	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por_hnf_slcway_partition0_rnf_vec1

Functions as the control register for RN-Fs that can allocate to partition 0 (ways 0, 1, 2, and 3).

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hCB0

The following image shows the higher register bit assignments.

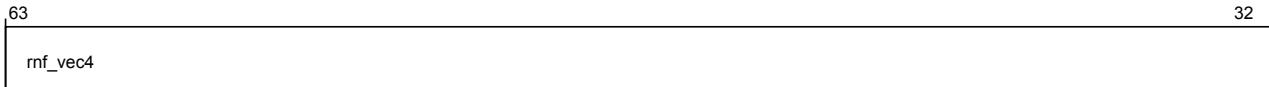


Figure 3-1262 por_hnf_por_hnf_slcway_partition0_rnf_vec1 (high)

The following table shows the port assignments for the HIFINetV2 chip.

Table 3-1282 por_hnf_por_hnf_slcway_partition0_rnf_vec1 (high)

Bits	Field name	Description	Type	Reset
63:32	rmf_vec4	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments.

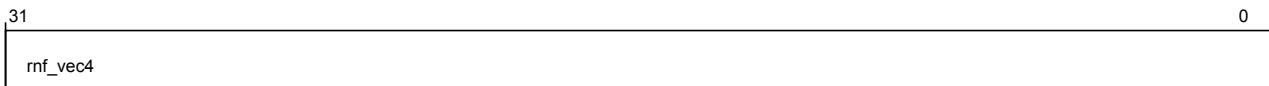


Figure 3-1263 por hnf por hnf slcway partition0 rnf vec1 (low)

The following table shows the port assignments for the SLCWAY partition0 module.

Table 3-1283 por hnf por hnf slcway partition0 rnf vec1 (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec4	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por hnf slcway partition1 rnf vec1

Functions as the control register for RN-Fs that can allocate to partition 1 (ways 4, 5, 6, and 7) for Logical RNF IDs 64 to 127.

Its characteristics are:

The following image shows the higher register bit assignments.

63 rnf_vec5 32

Figure 3-1264 por_hnf_por_hnf_slcway_partition1_rnf_vec1 (high)

The following table shows the por_hnf_slewway_partition1_rnf_vec1 higher register bit assignments.

Table 3-1284 por_hnf_por_hnf_slcway_partition1_rnf_vec1 (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_vec5	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments.

31 rnf_vec5 0

Figure 3-1265 por hnf por hnf slcway partition1 rnf vec1 (low)

The following table shows the port, hnf, slcway, partition1, rnf, vec1, lower register bit assignments.

Table 3-1285 por_hnf_por_hnf_slcway_partition1_rnf_vec1 (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec5	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por_hnf_slcway_partition2_rnf_vec1

Functions as the control register for RN-Fs that can allocate to partition 2 (ways 8, 9, 10, and 11) for Logical RNF IDs 64 to 127.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCC0

Usage constraints	Only accessible by secure accesses.
Secure group	por hnf secure register groups override.slc lock ways

override

Figure 3-1266 por hnf por hnf slcway partition2 rnf vec1 (high)

The following table shows the por_hnf_slewway_partition2_rnf_vec1 higher register bit assignments.

Table 3-1286 por_hnf_por_hnf_slcway_partition2_rnf_vec1 (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_vec6	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments.

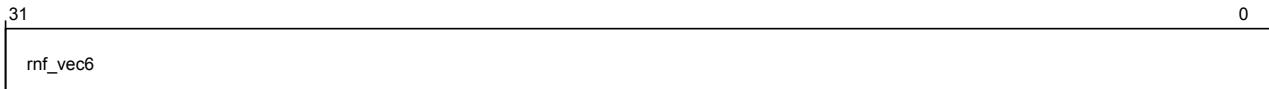


Figure 3-1267 por_hnf_por_hnf_slcway_partition2_rnf_vec1 (low)

The following table shows the port assignments for the RNF vector lower register bit assignments.

Table 3-1287 por_hnf_por_hnf_slcway_partition2_rnf_vec1 (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec6	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por_hnf_slcway_partition3_rnf_vec1

Functions as the control register for RN-Fs that can allocate to partition 3 (ways 12, 13, 14, and 15) for Logical RNF IDs 64 to 127.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCC8

Usage constraints	Only accessible by secure accesses.
Secure group	por hnf secure register groups override.slc lock ways

override

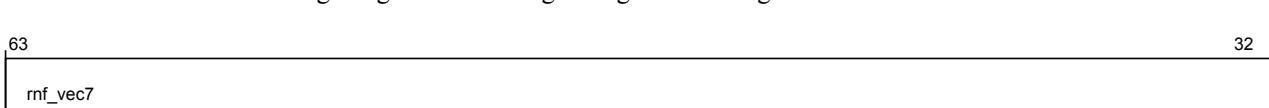


Figure 3-1268 por hnf por hnf slcway partition3 rnf vec1 (high)

The following table shows the por bnf slcway partition3 rnf vec1 higher register bit assignments

Table 3-1288 por hnf por hnf slcway partition3 rnf vec1 (high)

Bits	Field name	Description	Type	Reset
63:32	rnf_vec7	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

The following image shows the lower register bit assignments.

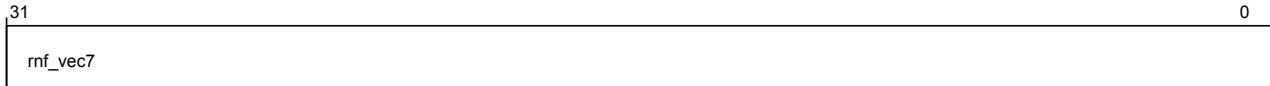


Figure 3-1269 por_hnf_por_hnf_slcway_partition3_rnf_vec1 (low)

The following table shows the por_hnf_slcway_partition3_rnf_vec1 lower register bit assignments.

Table 3-1289 por_hnf_por_hnf_slcway_partition3_rnf_vec1 (low)

Bits	Field name	Description	Type	Reset
31:0	rnf_vec7	Bit vector mask; identifies which logical IDs of the RN-F can allocate	RW	64'hFFFFFFFFFFFFFFF

por_hnf_slcway_partition0_rni_vec

Functions as the control register for RN-Is that can allocate to partition 0 (ways 0, 1, 2, and 3).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC68

Usage constraints	Only accessible by secure accesses.
Secure group	por_hnf_secure_register_groups_override.slc_lock_ways

override

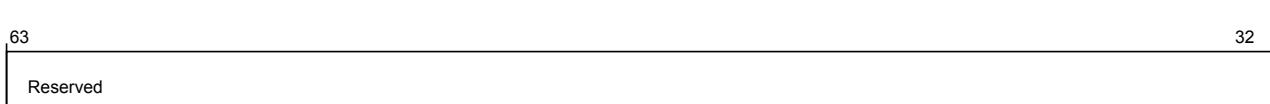


Figure 3-1270 por hnf por hnf slcway partition0 rpi vec (high)

The following table shows the port bit assignments for the `hbf_slewway_partition0_rni_vec` higher register bit assignments.

Table 3-1290 por hnf por hnf slcway partition0 rni vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments

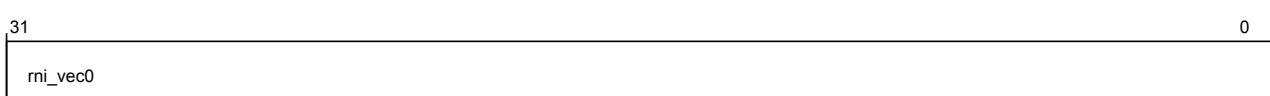


Figure 3-1271 por hnf por hnf slcway partition0 rni vec (low)

The following table shows the por_hnf_slcway_partition0_rni_vec lower register bit assignments.

Table 3-1291 por_hnf_por_hnf_slcway_partition0_rni_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rni_vec0	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFFF

por_hnf_slcway_partition1_rni_vec

Functions as the control register for RN-Is that can allocate to partition 1 (ways 4, 5, 6, and 7).

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC70
Register reset	64'b11111111111111111111111111111111
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

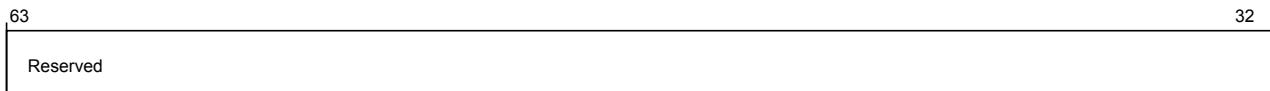


Figure 3-1272 por_hnf_por_hnf_slcway_partition1_rni_vec (high)

The following table shows the por_hnf_slcway_partition1_rni_vec higher register bit assignments.

Table 3-1292 por_hnf_por_hnf_slcway_partition1_rni_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

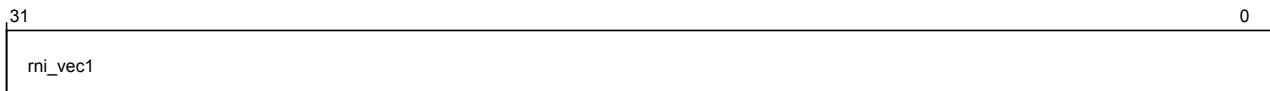


Figure 3-1273 por_hnf_por_hnf_slcway_partition1_rni_vec (low)

The following table shows the por_hnf_slcway_partition1_rni_vec lower register bit assignments.

Table 3-1293 por_hnf_por_hnf_slcway_partition1_rni_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rni_vec1	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFFF

por_hnf_slcway_partition2_rni_vec

Functions as the control register for RN-Is that can allocate to partition 2 (ways 8, 9, 10, and 11).

Its characteristics are:

The following image shows the higher register bit assignments.

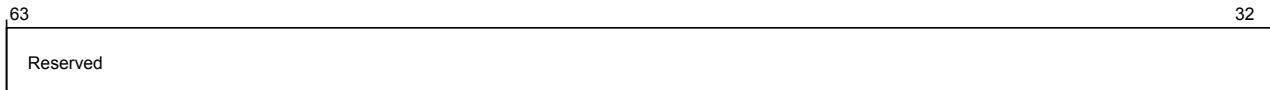


Figure 3-1274 por_hnf_por_hnf_slcway_partition2_rni_vec (high)

The following table shows the por_hnf_sleway_partition2_rni_vec higher register bit assignments.

Table 3-1294 por_hnf_por_hnf_slcway_partition2_rni_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

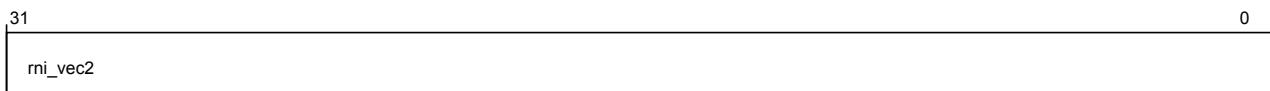


Figure 3-1275 por hnf por hnf slcway partition2 rni vec (low)

The following table shows the port pin assignments for the `hf_slewway_partition2` module.

Table 3-1295 por hnf por hnf slcway partition2 rni vec (low)

Bits	Field name	Description	Type	Reset
31:0	rni_vec2	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFFF

por_hnf_slcway_partition3_rni_vec

Functions as the control register for RN-Is that can allocate to partition 3 (ways 12, 13, 14, and 15).

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC80

The following image shows the higher register bit assignments.

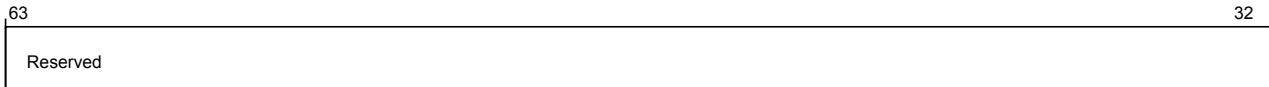


Figure 3-1276 por_hnf_por_hnf_slcway_partition3_rni_vec (high)

The following table shows the port assignments for the higher register bit assignments.

Table 3-1296 por_hnf_por_hnf_slcway_partition3_rni_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

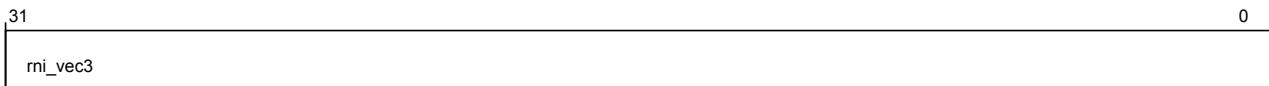


Figure 3-1277 por hnf por hnf slcway partition3 rni vec (low)

The following table shows the port pin assignments for the `slcway` partition3 memory vector lower register bit assignments.

Table 3-1297 por hnf por hnf slcway partition3 rni vec (low)

Bits	Field name	Description	Type	Reset
31:0	rni_vec3	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFFF

por hnf slcway partition0 rnd vec

Functions as the control register for RN-Ds that can allocate to partition 0 (ways 0, 1, 2, and 3).

Its characteristics are:

The following image shows the higher register bit assignments.

63	Reserved	32
----	----------	----

Figure 3-1278 por_hnf_por_hnf_slcway_partition0_rnd_vec (high)

The following table shows the por_hnf_slcway_partition0_rnd_vec higher register bit assignments.

Table 3-1298 por_hnf_por_hnf_slcway_partition0_rnd_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31	rnd_vec0	0
----	----------	---

Figure 3-1279 por_hnf_por_hnf_slcway_partition0_rnd_vec (low)

The following table shows the por_hnf_slcway_partition0_rnd_vec lower register bit assignments.

Table 3-1299 por_hnf_por_hnf_slcway_partition0_rnd_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnd_vec0	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFF

por_hnf_slcway_partition1_rnd_vec

Functions as the control register for RN-Ds that can allocate to partition 1 (ways 4, 5, 6, and 7).

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hC90
Register reset	64'b11111111111111111111111111111111
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

63	Reserved	32
----	----------	----

Figure 3-1280 por_hnf_por_hnf_slcway_partition1_rnd_vec (high)

The following table shows the por_hnf_slcway_partition1_rnd_vec higher register bit assignments.

Table 3-1300 por_hnf_por_hnf_slcway_partition1_rnd_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

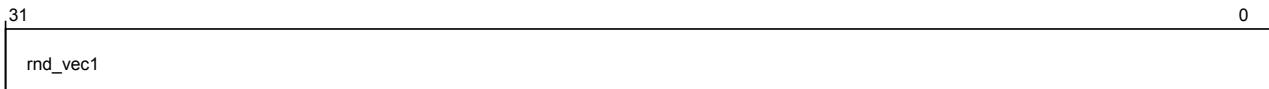


Figure 3-1281 por_hnf_por_hnf_slcway_partition1_rnd_vec (low)

The following table shows the por_hnf_slewway_partition1_rnd_vec lower register bit assignments.

Table 3-1301 por_hnf_por_hnf_slcway_partition1_rnd_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnd_vec1	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFFF

por_hnf_slcway_partition2_rnd_vec

Functions as the control register for RN-Ds that can allocate to partition 2 (ways 8, 9, 10, and 11).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hC98

Usage constraints	Only accessible by secure accesses.
Secure group	por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher resisted hit assignments.

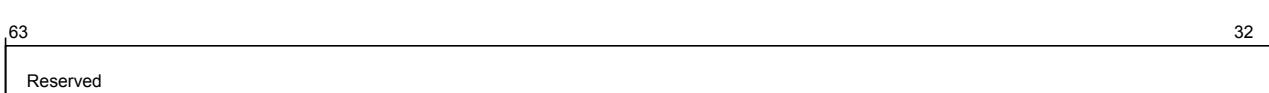


Figure 3-1282 nor_hnf nor_hnf_slcway partition? rnd_vec (high)

The following table shows the nor_hnf_slcway_partition2_rnd_vec higher register bit assignments.

Table 3-1302 nor hpf nor hpf slcway partition? rnd vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments

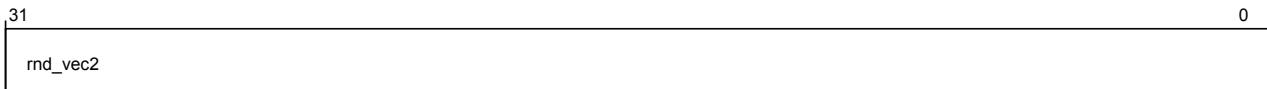


Figure 3-1283 por_hnf_por_hnf_slcway_partition2_rnd_vec (low)

The following table shows the por_hnf_slcway_partition2_rnd_vec lower register bit assignments.

Table 3-1303 por_hnf_por_hnf_slcway_partition2_rnd_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnd_vec2	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFF

por_hnf_slcway_partition3_rnd_vec

Functions as the control register for RN-Ds that can allocate to partition 3 (ways 12, 13, 14, and 15).

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCA0

Register reset 64'b11111111111111111111111111111111

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

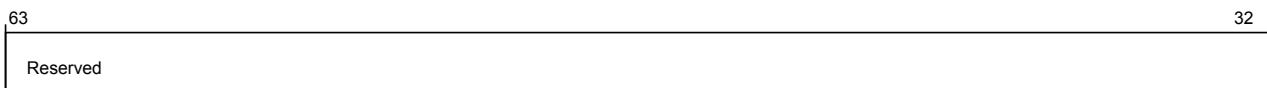


Figure 3-1284 por_hnf_por_hnf_slcway_partition3_rnd_vec (high)

The following table shows the por_hnf_slcway_partition3_rnd_vec higher register bit assignments.

Table 3-1304 por_hnf_por_hnf_slcway_partition3_rnd_vec (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

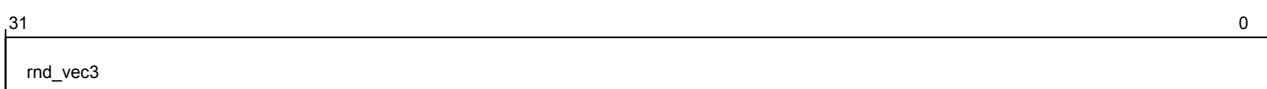


Figure 3-1285 por_hnf_por_hnf_slcway_partition3_rnd_vec (low)

The following table shows the por_hnf_slcway_partition3_rnd_vec lower register bit assignments.

Table 3-1305 por_hnf_por_hnf_slcway_partition3_rnd_vec (low)

Bits	Field name	Description	Type	Reset
31:0	rnd_vec3	Bit vector mask; identifies which logical IDs of the RN-I/RN-D can allocate	RW	32'hFFFFFF

por_hnf_rn_region_lock

Functions as the enable register for source-based SLC way allocation.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCA8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slc_lock_ways

The following image shows the higher register bit assignments.

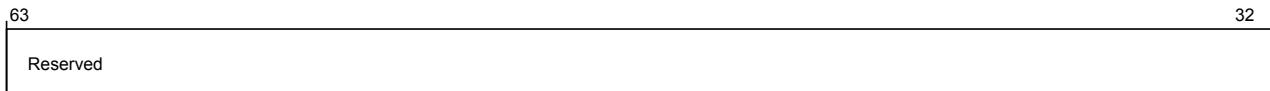


Figure 3-1286 por_hnf_por_hnf_rn_region_lock (high)

The following table shows the por_hnf_rn_region_lock higher register bit assignments.

Table 3-1306 por_hnf_por_hnf_rn_region_lock (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

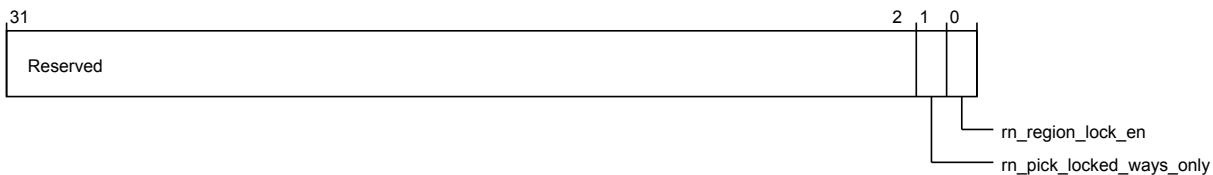


Figure 3-1287 por_hnf_por_hnf_rn_region_lock (low)

The following table shows the por_hnf_rn_region_lock lower register bit assignments.

Table 3-1307 por_hnf_por_hnf_rn_region_lock (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1	rn_pick_locked_ways_only	Specifies which ways the programmed RNs can allocate new cache lines to 1'b0: Programmed RN will choose all ways including locked 1'b1: Programmed RN will only allocate in locked ways	RW	1'b0
0	rn_region_lock_en	Enables SRC-based region locking 1'b0: SRC based way locking is disabled 1'b1: SRC based way locking is enabled	RW	1'b0

por_hnf_sf_cxg_blocked_ways

Specifies the SF ways that are blocked for remote chip to use in CML mode.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCD0

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

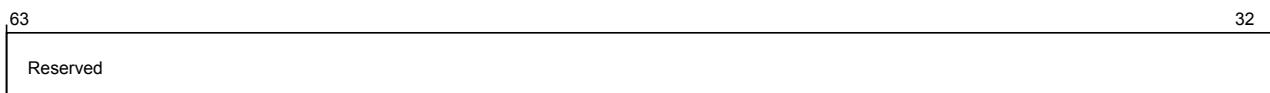


Figure 3-1288 por_hnf_por_hnf_sf_cxg_blocked_ways (high)

The following table shows the por_hnf_sf_cxg_blocked_ways higher register bit assignments.

Table 3-1308 por_hnf_por_hnf_sf_cxg_blocked_ways (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

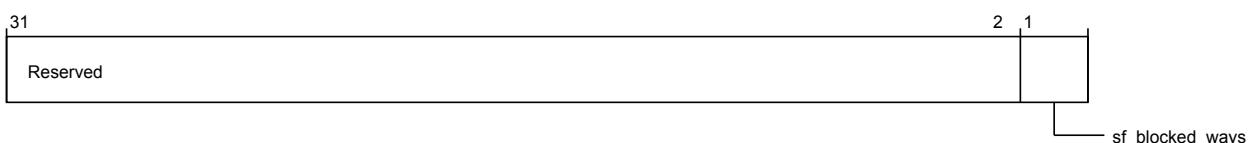


Figure 3-1289 por_hnf_por_hnf_sf_cxg_blocked_ways (low)

The following table shows the por_hnf_sf_cxg_blocked_ways lower register bit assignments.

Table 3-1309 por_hnf_por_hnf_sf_cxg_blocked_ways (low)

Bits	Field name	Description	Type	Reset
31:2	Reserved	Reserved	RO	-
1:0	sf_blocked_ways	<p>Number of SF ways blocked for remote chips to use in CML mode (0%, 25%, 50%, or 75%)</p> <p>2'b00: No ways are blocked; all SF ways could be used by local or remote RN-Fs</p> <p>2'b01: Lower 25% ways are blocked from remote RN-Fs.</p> <p>16 way SF: ways 3:0 for local RN-Fs only; ways 15:4 for local and remote RN-Fs</p> <p>32 way SF: ways 7:0 for locak RN-Fs only; ways 31:8 for locan and remote RN-Fs</p> <p>2'b10: Lower 50% ways are blocked from remote RN-Fs.</p> <p>16 way SF: ways 7:0 for local RN-Fs only; ways 15:8 for local and remote RN-Fs</p> <p>32 way SF: ways 15:0 for locak RN-Fs only; ways 31:16 for locan and remote RN-Fs</p> <p>2'b11: Lower 75% ways are blocked from remote RN-Fs.</p> <p>16 way SF: ways 11:0 for local RN-Fs only; ways 15:12 for local and remote RN-Fs</p> <p>32 way SF: ways 23:0 for locak RN-Fs only; ways 31:24 for locan and remote RN-Fs</p>	RW	2'b00

por_hnf_cxg_ha_metadata_exclusion_list

Functions as the control register to identify CXG HA which does not support metadata

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hCE0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

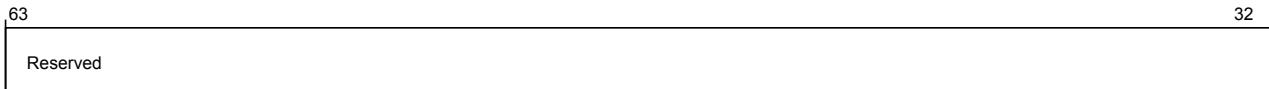


Figure 3-1290 por_hnf_por_hnf_cxg_ha_metadata_exclusion_list (high)

The following table shows the por_hnf_cxg_ha_metadata_exclusion_list higher register bit assignments.

Table 3-1310 por_hnf_por_hnf_cxg_ha_metadata_exclusion_list (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31	cxg_ha_vec	0
----	------------	---

Figure 3-1291 por_hnf_por_hnf_cxg_ha_metadata_exclusion_list (low)

The following table shows the por_hnf_cxg_ha_metadata_exclusion_list lower register bit assignments.

Table 3-1311 por_hnf_por_hnf_cxg_ha_metadata_exclusion_list (low)

Bits	Field name	Description	Type	Reset
31:0	cxg_ha_vec	Bit vector mask; identifies which logical IDs of the CXG HA does not support metadata	RW	32'h00000000

por_hnf_cxg_ha_smp_exclusion_list

Functions as the control register to identify CXG HA not connected to SMP CCIX link

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCD8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

63	Reserved	32
----	----------	----

Figure 3-1292 por_hnf_por_hnf_cxg_ha_smp_exclusion_list (high)

The following table shows the por_hnf_cxg_ha_smp_exclusion_list higher register bit assignments.

Table 3-1312 por_hnf_por_hnf_cxg_ha_smp_exclusion_list (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31	cxg_ha_vec	0
----	------------	---

Figure 3-1293 por_hnf_por_hnf_cxg_ha_smp_exclusion_list (low)

The following table shows the por_hnf_cxg_ha_smp_exclusion_list lower register bit assignments.

Table 3-1313 por_hnf_por_hnf_cxg_ha_smp_exclusion_list (low)

Bits	Field name	Description	Type	Reset
31:0	cxg_ha_vec	Bit vector mask; identifies which logical IDs of the CXG HA does not connect to SMP CCIX link	RW	32'h00000000

hn_sam_hash_addr_mask_reg

Configures the address mask that is applied before hashing the address bits.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCF0

Register reset	64 bits
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

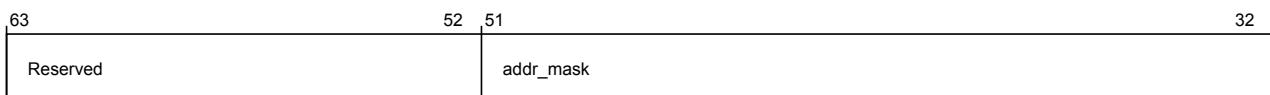


Figure 3-1294 por_hnf_hn_sam_hash_addr_mask_reg (high)

The following table shows the hn sam hash addr mask reg higher register bit assignments.

Table 3-1314 por hnf hn sam hash addr mask req (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	Address mask applied before hashing	RW	46'h3FFFFFFFFFFF

The following image shows the lower register bit assignments.

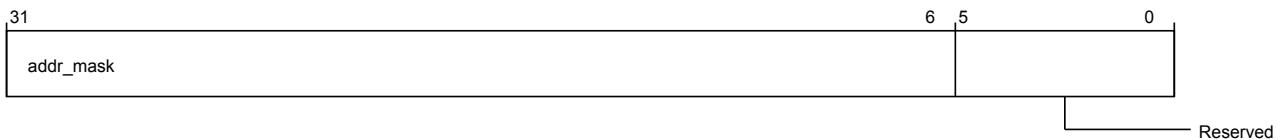


Figure 3-1295 por_hnf_hn_sam_hash_addr_mask_reg (low)

The following table shows the hn_sam_hash_addr_mask_reg lower register bit assignments.

Table 3-1315 por_hnf_hn_sam_hash_addr_mask_reg (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask	Address mask applied before hashing	RW	46'h3FFFFFFFFF
5:0	Reserved	Reserved	RO	-

hn_sam_region_cmp_addr_mask_reg

Configures the address mask that is applied before memory region compare.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCF8

Register reset 64'b11111111111111111111111111111111

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device. and This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

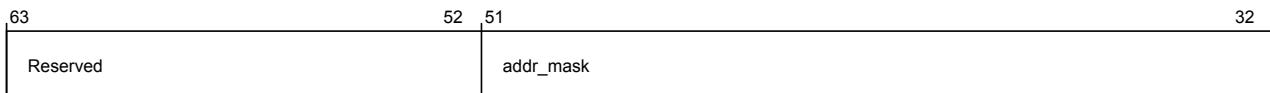


Figure 3-1296 por_hnf_hn_sam_region_cmp_addr_mask_reg (high)

The following table shows the hn_sam_region_cmp_addr_mask_reg higher register bit assignments.

Table 3-1316 por_hnf_hn_sam_region_cmp_addr_mask_reg (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	Address mask applied before memory region compare	RW	32'hFFFFFF

The following image shows the lower register bit assignments.



Figure 3-1297 por_hnf_hn_sam_region_cmp_addr_mask_reg (low)

The following table shows the hn_sam_region_cmp_addr_mask_reg lower register bit assignments.

Table 3-1317 por_hnf_hn_sam_region_cmp_addr_mask_reg (low)

Bits	Field name	Description	Type	Reset
31:20	addr_mask	Address mask applied before memory region compare	RW	32'hFFFFFF
19:0	Reserved	Reserved	RO	-

por_hnf_sam_control

Configures HN-F SAM. All top_address_bit fields must be between bits 47 and 28 of the address. top_address_bit2 > top_address_bit1 > top_address_bit0. Must be configured to match corresponding por_rnsam_sys_cache_grp_sn_sam_cfgN register in the RN SAM.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD00

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_hnf_secure_register_groups_override.sam_control

The following is a sample of a high quality document:

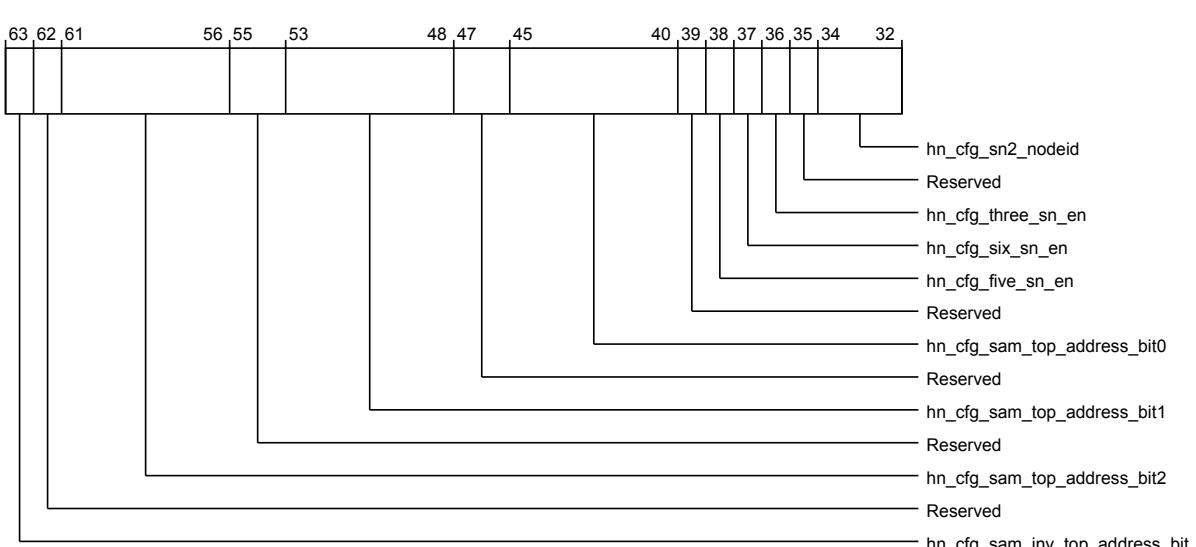


Figure 3-1298 por hnf por hnf sam control (high)

The following table shows the por_hnf_sam control higher register bit assignments.

Table 3-1318 por_hnf_por_hnf_sam_control (high)

Bits	Field name	Description	Type	Reset
63	hn_cfg_sam_inv_top_address_bit	Inverts the top address bit (hn_cfg_sam_top_address_bit1 if 3-SN, hn_cfg_sam_top_address_bit2 if 6-SN) NOTE: Can only be used when the address map does not have unique address bit combinations.	RW	1'h0
62	Reserved	Reserved	RO	-
61:56	hn_cfg_sam_top_address_bit2	Bit position of top_address_bit2; used for address hashing in 6-SN configuration	RW	6'h00
55:54	Reserved	Reserved	RO	-
53:48	hn_cfg_sam_top_address_bit1	Bit position of top_address_bit1; used for address hashing in 3-SN/6-SN configuration	RW	6'h00
47:46	Reserved	Reserved	RO	-
45:40	hn_cfg_sam_top_address_bit0	Bit position of top_address_bit0; used for address hashing in 3-SN/6-SN configuration	RW	6'h00
39	Reserved	Reserved	RO	-
38	hn_cfg_five_sn_en	Enables 5-SN configuration	RW	1'b0
37	hn_cfg_six_sn_en	Enables 6-SN configuration	RW	1'b0
36	hn_cfg_three_sn_en	Enables 3-SN configuration	RW	1'b0
35	Reserved	Reserved	RO	-
34:32	hn_cfg_sn2_nodeid	SN 2 node ID	RW	11'h0

The following image shows the lower register bit assignments.

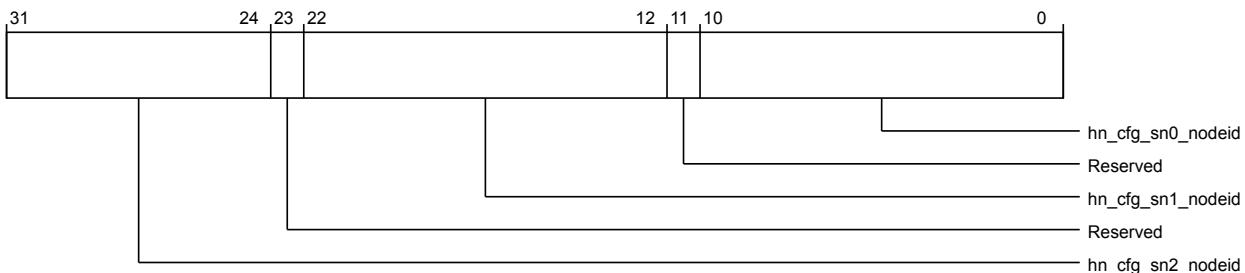


Figure 3-1299 por_hnf_por_hnf_sam_control (low)

The following table shows the por_hnf_sam_control lower register bit assignments.

Table 3-1319 por_hnf_por_hnf_sam_control (low)

Bits	Field name	Description	Type	Reset
31:24	hn_cfg_sn2_nodeid	SN 2 node ID	RW	11'h0
23	Reserved	Reserved	RO	-
22:12	hn_cfg_sn1_nodeid	SN 1 node ID	RW	11'h0
11	Reserved	Reserved	RO	-
10:0	hn_cfg_sn0_nodeid	SN 0 node ID	RW	11'h0

por_hnf_sam_memregion0

Configures range-based memory region 0 in HN-F SAM.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD08

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

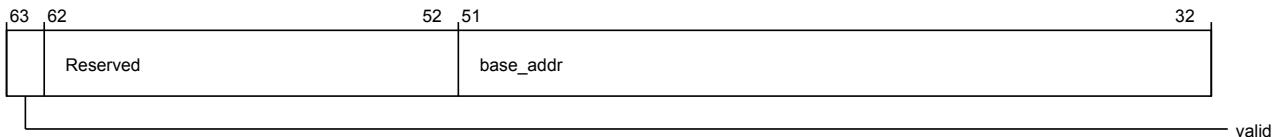


Figure 3-1300 por_hnf_por_hnf_sam_memregion0 (high)

The following table shows the por_hnf_sam_memregion0 higher register bit assignments.

Table 3-1320 por_hnf_por_hnf_sam_memregion0 (high)

Bits	Field name	Description	Type	Reset
63	valid	Memory region 0 valid 1'b0: Not valid 1'b1: Valid for memory region comparison	RW	1'h0
62:52	Reserved	Reserved	RO	-
51:32	base_addr	Base address of memory region 0 CONSTRAINT: Must be an integer multiple of region size.	RW	32'h0

The following image shows the lower register bit assignments.

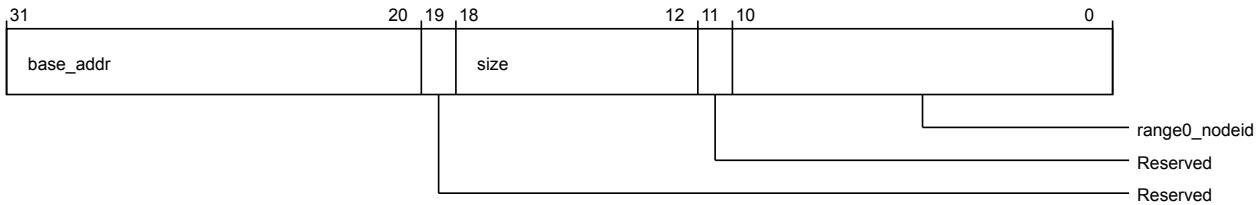


Figure 3-1301 por_hnf_por_hnf_sam_memregion0 (low)

The following table shows the por_hnf_sam_memregion0 lower register bit assignments.

Table 3-1321 por_hnf_por_hnf_sam_memregion0 (low)

Bits	Field name	Description	Type	Reset
31:20	base_addr	Base address of memory region 0 CONSTRAINT: Must be an integer multiple of region size.	RW	32'h0
19	Reserved	Reserved	RO	-
18:12	size	Memory region 0 size CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).	RW	7'h0
11	Reserved	Reserved	RO	-
10:0	range0_nodeid	Memory region 0 target node ID	RW	11'h0

por_hnf_sam_memregion0_end_addr

Configures end address memory region 0 in HN-F SAM.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD38
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

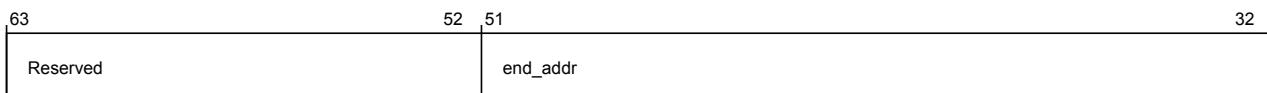


Figure 3-1302 por_hnf_por_hnf_sam_memregion0_end_addr (high)

The following table shows the por_hnf_sam_memregion0_end_addr higher register bit assignments.

Table 3-1322 por_hnf_por_hnf_sam_memregion0_end_addr (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	end_addr	End address of memory region 0	RW	32'h0

The following image shows the lower register bit assignments.



Figure 3-1303 por_hnf_por_hnf_sam_memregion0_end_addr (low)

The following table shows the por_hnf_sam_memregion0_end_addr lower register bit assignments.

Table 3-1323 por_hnf_por_hnf_sam_memregion0_end_addr (low)

Bits	Field name	Description	Type	Reset
31:20	end_addr	End address of memory region 0	RW	32'h0
19:0	Reserved	Reserved	RO	-

por_hnf_sam_memregion1

Configures range-based memory region 1 in HN-F SAM.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD10

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

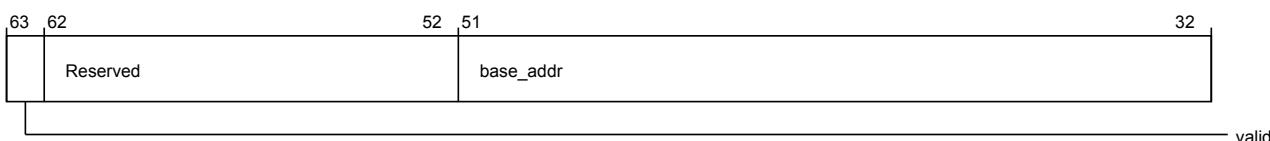


Figure 3-1304 por_hnf_por_hnf_sam_memregion1 (high)

The following table shows the por_hnf_sam_memregion1 higher register bit assignments.

Table 3-1324 por_hnf_por_hnf_sam_memregion1 (high)

Bits	Field name	Description	Type	Reset
63	valid	Memory region 1 valid 1'b0: Not valid 1'b1: Valid for memory region comparison	RW	1'h0
62:52	Reserved	Reserved	RO	-
51:32	base_addr	Base address of memory region 1 CONSTRAINT: Must be an integer multiple of region size.	RW	32'h0

The following image shows the lower register bit assignments.

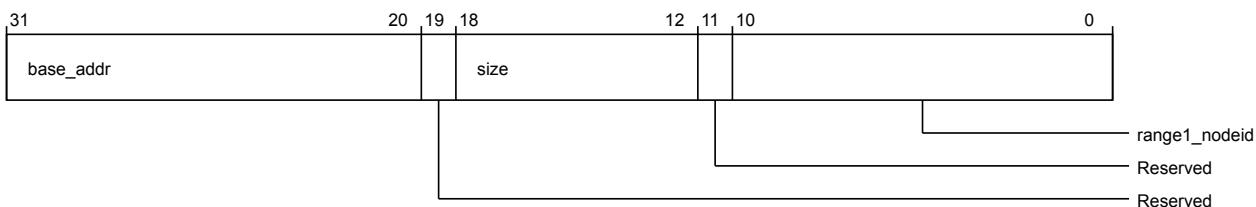


Figure 3-1305 por_hnf_por_hnf_sam_memregion1 (low)

The following table shows the por_hnf_sam_memregion1 lower register bit assignments.

Table 3-1325 por_hnf_por_hnf_sam_memregion1 (low)

Bits	Field name	Description	Type	Reset
31:20	base_addr	Base address of memory region 1 CONSTRAINT: Must be an integer multiple of region size.	RW	32'h0
19	Reserved	Reserved	RO	-
18:12	size	Memory region 1 size CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).	RW	7'h0
11	Reserved	Reserved	RO	-
10:0	range1_nodeid	Memory region 1 target node ID	RW	11'h0

por_hnf_sam_memregion1_end_addr

Configures end address memory region 1 in HN-F SAM.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hD40

Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

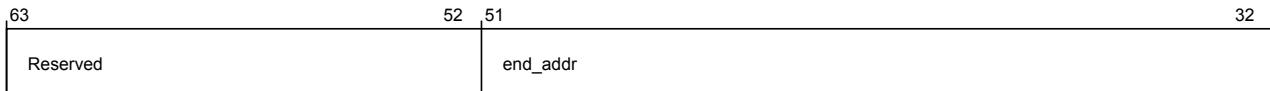


Figure 3-1306 por_hnf_por_hnf_sam_memregion1_end_addr (high)

The following table shows the por_hnf_sam_memregion1_end_addr higher register bit assignments.

Table 3-1326 por_hnf_por_hnf_sam_memregion1_end_addr (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	end_addr	End address of memory region 1	RW	32'h0

The following image shows the lower register bit assignments.

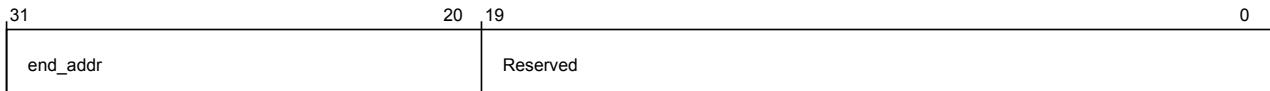


Figure 3-1307 por_hnf_por_hnf_sam_memregion1_end_addr (low)

The following table shows the por_hnf_sam_memregion1_end_addr lower register bit assignments.

Table 3-1327 por_hnf_por_hnf_sam_memregion1_end_addr (low)

Bits	Field name	Description	Type	Reset
31:20	end_addr	End address of memory region 1	RW	32'h0
19:0	Reserved	Reserved	RO	-

por_hnf_sam_sn_properties

Configures properties for all six SN targets and two range-based SN targets.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD18
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

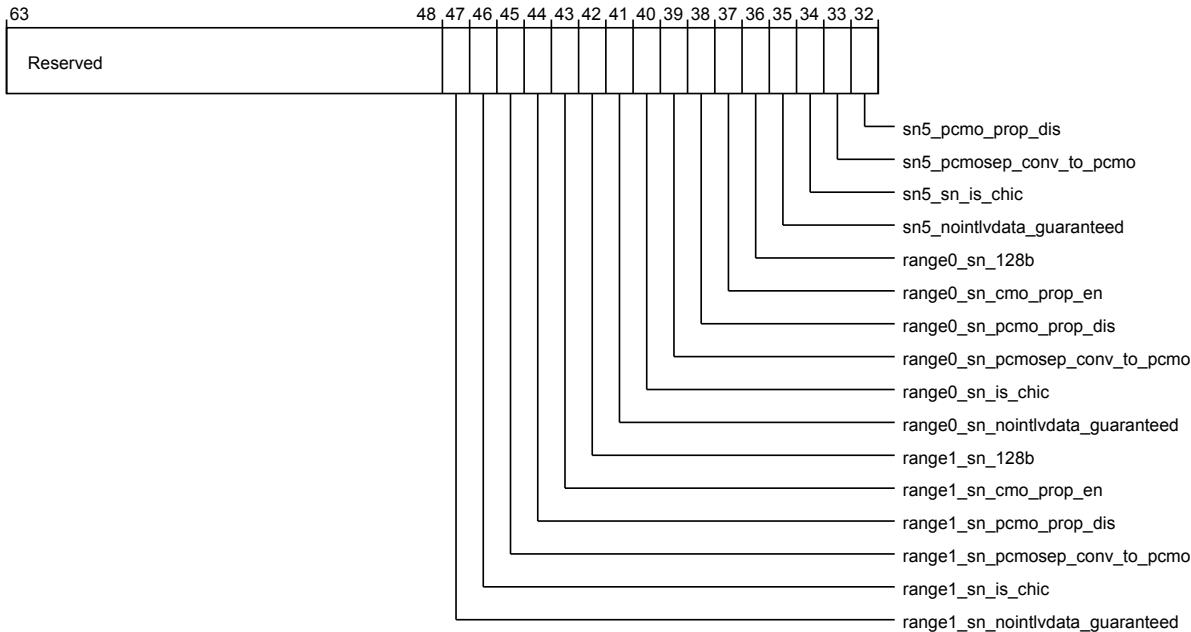


Figure 3-1308 por_hnf_por_hnf_sam_sn_properties (high)

The following table shows the por_hnf_sam_sn_properties higher register bit assignments.

Table 3-1328 por_hnf_por_hnf_sam_sn_properties (high)

Bits	Field name	Description	Type	Reset
63:48	Reserved	Reserved	RO	-
47	range1_sn_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
46	range1_sn_is_chic	Indicates that the range 1 SN is a CHI-C SN when set	RW	1'b0
45	range1_sn_pcosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for range 1 SN when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
44	range1_sn_pcmo_prop_dis	Disables PCMO (persistent CMO) propagation for range 1 SN when set	RW	1'b0
43	range1_sn_cmo_prop_en	Enables CMO propagation for range 1 SN	RW	1'b0
42	range1_sn_128b	Data width of range 1 SN 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
41	range0_sn_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
40	range0_sn_is_chic	Indicates that the range 0 SN is a CHI-C SN when set	RW	1'b0

Table 3-1328 por_hnf_por_hnf_sam_sn_properties (high) (continued)

Bits	Field name	Description	Type	Reset
39	range0_sn_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for range 0 SN when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
38	range0_sn_pcmo_prop_dis	Disables PCMO (persistent CMO) propagation for range 0 SN when set	RW	1'b0
37	range0_sn_cmo_prop_en	Enables CMO propagation for range 0 SN	RW	1'b0
36	range0_sn_128b	Data width of range 0 SN 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
35	sn5_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
34	sn5_sn_is_chic	Indicates that SN5 is a CHI-C SN when set	RW	1'b0
33	sn5_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN 5 when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
32	sn5_pcmo_prop_dis	Disables PCMO propagation for SN 5 when set	RW	1'b0

The following image shows the lower register bit assignments.

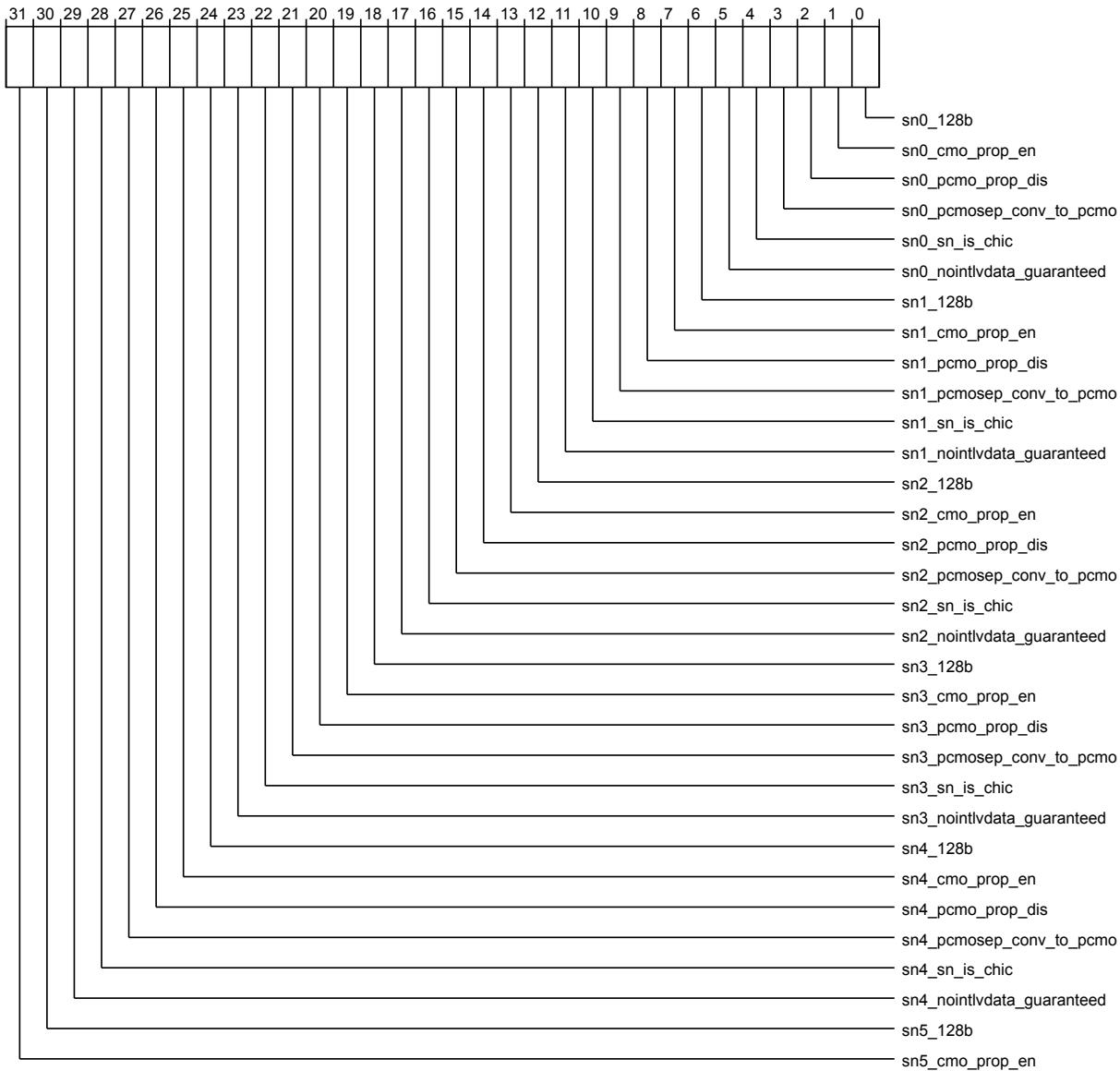


Figure 3-1309 por_hnf_por_hnf_sam_sn_properties (low)

The following table shows the por_hnf_sam_sn_properties lower register bit assignments.

Table 3-1329 por_hnf_por_hnf_sam_sn_properties (low)

Bits	Field name	Description	Type	Reset
31	sn5_cmo_prop_en	Enables CMO propagation for SN 5 when set	RW	1'b0
30	sn5_128b	Data width of SN 5 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
29	sn4_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
28	sn4_sn_is_chic	Indicates that SN4 is a CHI-C SN when set	RW	1'b0

Table 3-1329 por_hnf_por_hnf_sam_sn_properties (low) (continued)

Bits	Field name	Description	Type	Reset
27	sn4_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN 4 when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
26	sn4_pcmo_prop_dis	Disables PCMO propagation for SN 4 when set	RW	1'b0
25	sn4_cmo_prop_en	Enables CMO propagation for SN 4 when set	RW	1'b0
24	sn4_128b	Data width of SN 4 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
23	sn3_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
22	sn3_sn_is_chic	Indicates that SN3 is a CHI-C SN when set	RW	1'b0
21	sn3_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN 3 when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
20	sn3_pcmo_prop_dis	Disables PCMO propagation for SN 3 when set	RW	1'b0
19	sn3_cmo_prop_en	Enables CMO propagation for SN 3 when set	RW	1'b0
18	sn3_128b	Data width of SN 3 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
17	sn2_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
16	sn2_sn_is_chic	Indicates that SN2 is a CHI-C SN when set	RW	1'b0
15	sn2_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN 2 when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
14	sn2_pcmo_prop_dis	Disables PCMO propagation for SN 2 when set	RW	1'b0
13	sn2_cmo_prop_en	Enables CMO propagation for SN 2 when set	RW	1'b0
12	sn2_128b	Data width of SN 2 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
11	sn1_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
10	sn1_sn_is_chic	Indicates that SN1 is a CHI-C SN when set	RW	1'b0
9	sn1_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN 1 when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
8	sn1_pcmo_prop_dis	Disables PCMO propagation for SN 1 when set	RW	1'b0

Table 3-1329 por_hnf_por_hnf_sam_sn_properties (low) (continued)

Bits	Field name	Description	Type	Reset
7	sn1_cmo_prop_en	Enables CMO propagation for SN 1 when set	RW	1'b0
6	sn1_128b	Data width of SN 1 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
5	sn0_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
4	sn0_sn_is_chic	Indicates that SN0 is a CHI-C SN when set	RW	1'b0
3	sn0_pcmodsep_conv_to_pcmod	Convert CleanSharedPersistSep to CleanSharedPersist for SN 0 when set CONSTRAINT: Should not be enabled when sn_pcmod_prop_dis bit is set to 1	RW	1'b0
2	sn0_pcmod_prop_dis	Disables PCMO propagation for SN 0 when set	RW	1'b0
1	sn0_cmo_prop_en	Enables CMO propagation for SN 0 when set	RW	1'b0
0	sn0_128b	Data width of SN 0 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0

por_hnf_sam_6sn_nodeid

Configures node IDs for slave nodes 3 to 5 in 6-SN configuration mode.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD20
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

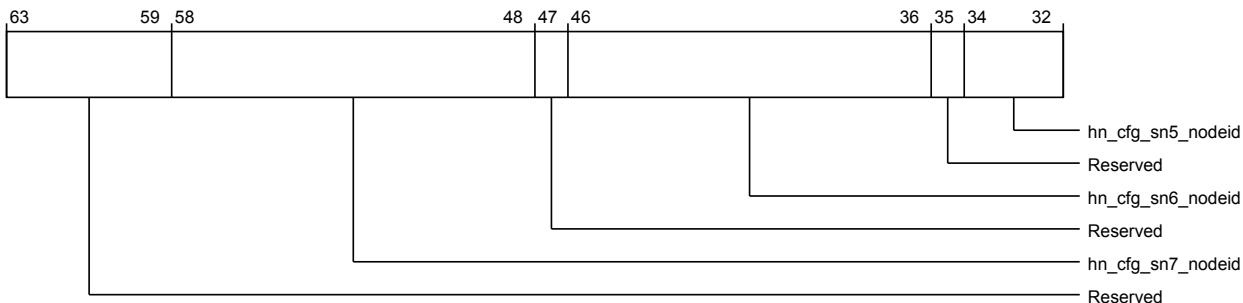


Figure 3-1310 por_hnf_por_hnf_sam_6sn_nodeid (high)

The following table shows the por_hnf_sam_6sn_nodeid higher register bit assignments.

Table 3-1330 por_hnf_por_hnf_sam_6sn_nodeid (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	hn_cfg_sn7_nodeid	SN 7 node ID	RW	11'h0
47	Reserved	Reserved	RO	-
46:36	hn_cfg_sn6_nodeid	SN 6 node ID	RW	11'h0
35	Reserved	Reserved	RO	-
34:32	hn_cfg_sn5_nodeid	SN 5 node ID	RW	11'h0

The following image shows the lower register bit assignments.

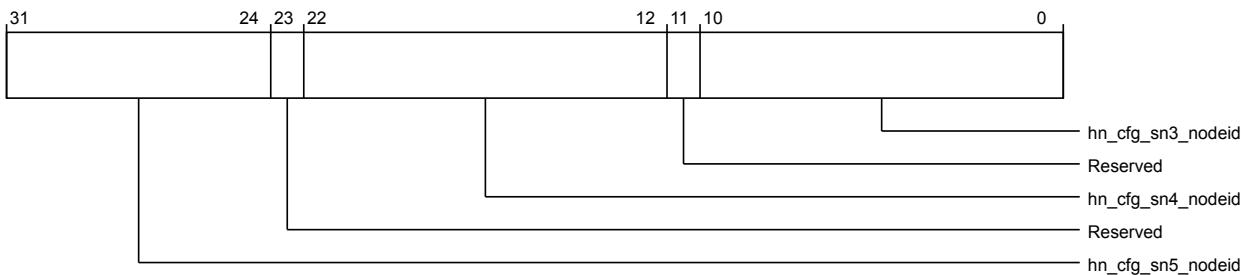


Figure 3-1311 por_hnf_por_hnf_sam_6sn_nodeid (low)

The following table shows the por_hnf_sam_6sn_nodeid lower register bit assignments.

Table 3-1331 por_hnf_por_hnf_sam_6sn_nodeid (low)

Bits	Field name	Description	Type	Reset
31:24	hn_cfg_sn5_nodeid	SN 5 node ID	RW	11'h0
23	Reserved	Reserved	RO	-
22:12	hn_cfg_sn4_nodeid	SN 4 node ID	RW	11'h0
11	Reserved	Reserved	RO	-
10:0	hn_cfg_sn3_nodeid	SN 3 node ID	RW	11'h0

por_hnf_sam_sn_properties1

Configures additional properties for all six SN targets and two range-based SN targets.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hCE8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

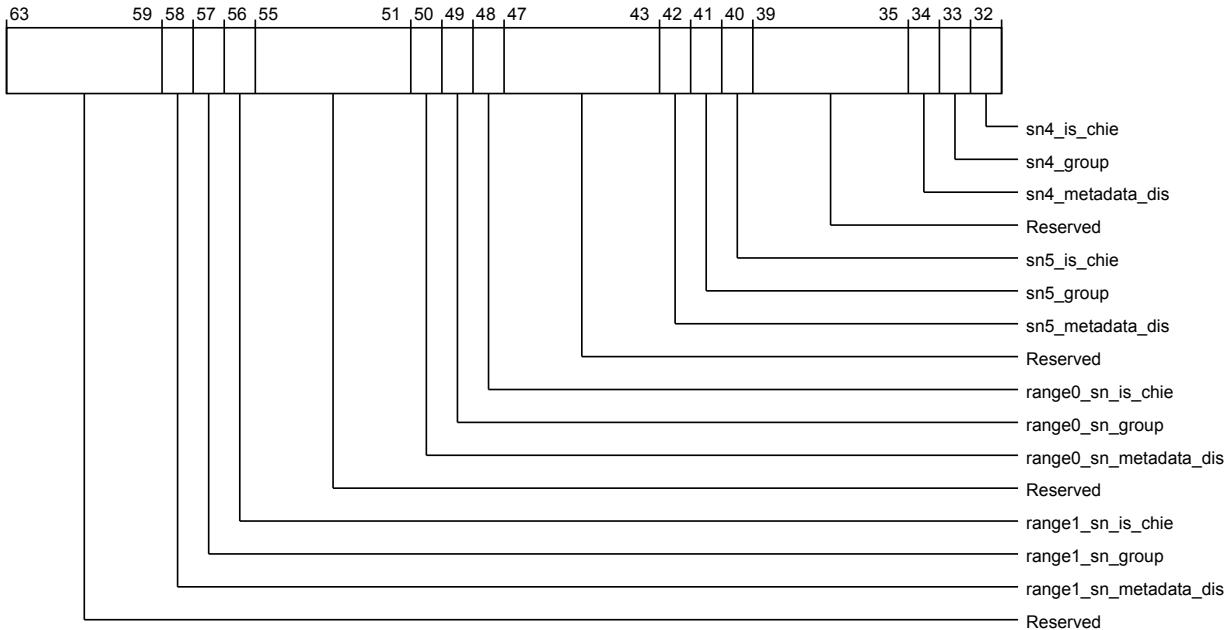


Figure 3-1312 por_hnf_por_hnf_sam_sn_properties1 (high)

The following table shows the por_hnf_sam_sn_properties1 higher register bit assignments.

Table 3-1332 por_hnf_por_hnf_sam_sn_properties1 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58	range1_sn_metadata_dis	HNF implements metadata termination flow for Range 1 SN when set	RW	1'b0
57	range1_sn_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
56	range1_sn_is_chie	Range 1 SN supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
55:51	Reserved	Reserved	RO	-
50	range0_sn_metadata_dis	HNF implements metadata termination flow for Range 0 SN when set	RW	1'b0
49	range0_sn_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
48	range0_sn_is_chie	Range 0 SN supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
47:43	Reserved	Reserved	RO	-

Table 3-1332 por_hnf_por_hnf_sam_sn_properties1 (high) (continued)

Bits	Field name	Description	Type	Reset
42	sn5_metadata_dis	HNF implements metadata termination flow for SN 5 when set	RW	1'b0
41	sn5_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
40	sn5_is_chie	SN 5 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
39:35	Reserved	Reserved	RO	-
34	sn4_metadata_dis	HNF implements metadata termination flow for SN 4 when set	RW	1'b0
33	sn4_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
32	sn4_is_chie	SN 4 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0

The following image shows the lower register bit assignments.

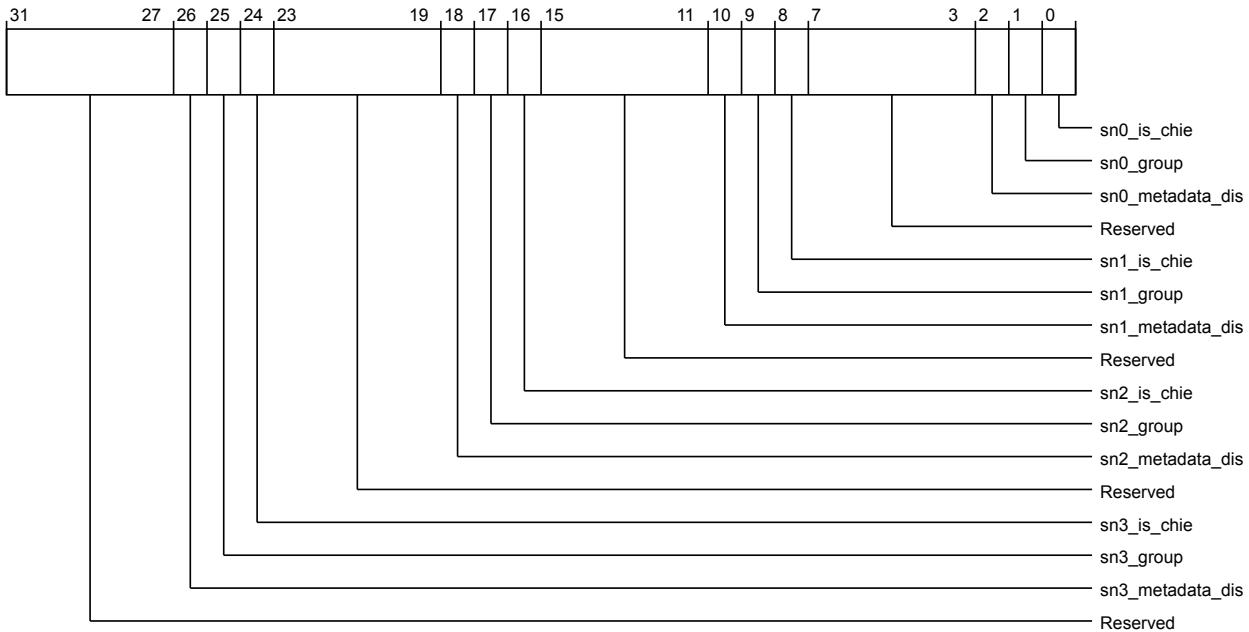


Figure 3-1313 por_hnf_por_hnf_sam_sn_properties1 (low)

The following table shows the por hnf sam sn properties1 lower register bit assignments.

Table 3-1333 por_hnf_por_hnf_sam_sn_properties1 (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26	sn3_metadata_dis	HNF implements metadata termination flow for SN 3 when set	RW	1'b0
25	sn3_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
24	sn3_is_chie	SN 3 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
23:19	Reserved	Reserved	RO	-
18	sn2_metadata_dis	HNF implements metadata termination flow for SN 2 when set	RW	1'b0
17	sn2_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
16	sn2_is_chie	SN 2 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
15:11	Reserved	Reserved	RO	-
10	sn1_metadata_dis	HNF implements metadata termination flow for SN 1 when set	RW	1'b0
9	sn1_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
8	sn1_is_chie	SN 1 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
7:3	Reserved	Reserved	RO	-
2	sn0_metadata_dis	HNF implements metadata termination flow for SN 0 when set	RW	1'b0
1	sn0_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
0	sn0_is_chie	SN 0 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0

por_hnf_sam_sn_properties2

Configures properties for SN-7 and SN-8.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hD30
Register reset	64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.



Figure 3-1314 por_hnf_por_hnf_sam_sn_properties2 (high)

The following table shows the por_hnf_sam_sn_properties2 higher register bit assignments.

Table 3-1334 por_hnf_por_hnf_sam_sn_properties2 (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

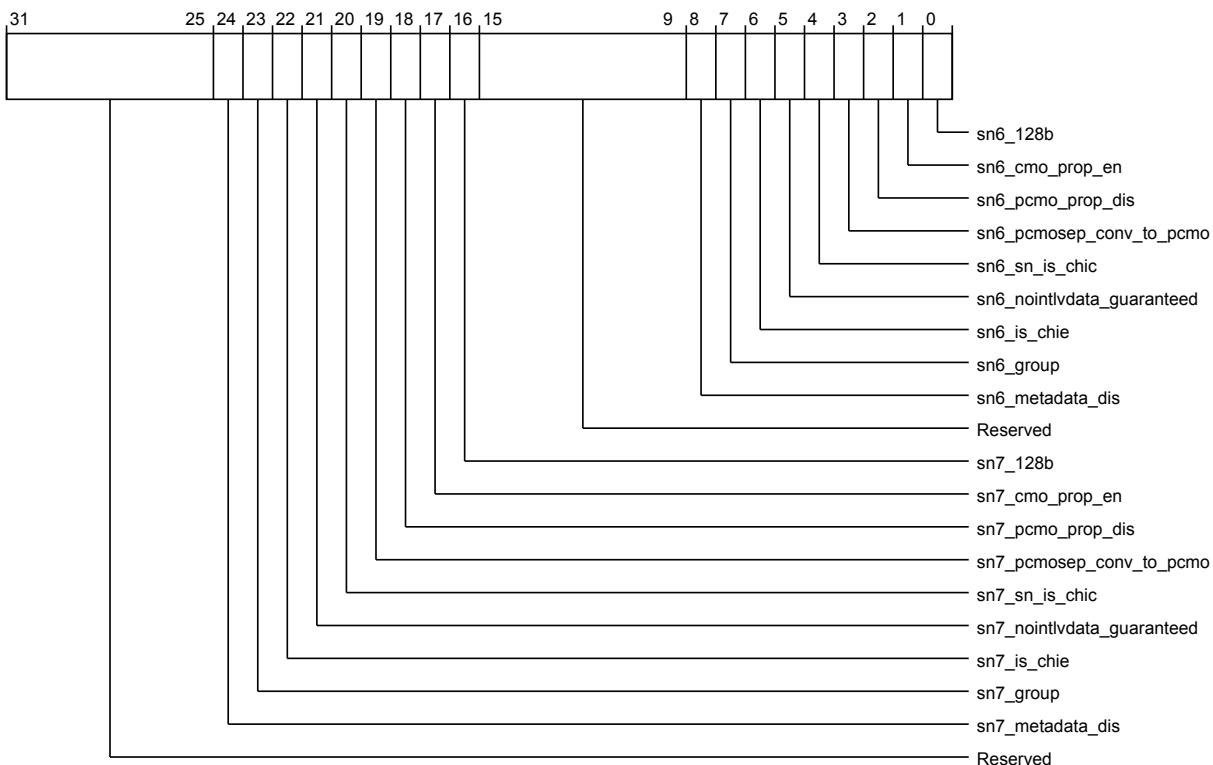


Figure 3-1315 por_hnf_por_hnf_sam_sn_properties2 (low)

The following table shows the por_hnf_sam_sn_properties2 lower register bit assignments.

Table 3-1335 por_hnf_por_hnf_sam_sn_properties2 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	sn7_metadata_dis	HNF implements metadata termination flow for SN 7 when set	RW	1'b0
23	sn7_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
22	sn7_is_chie	SN 7 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
21	sn7_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
20	sn7_sn_is_chic	Indicates that SN7 is a CHI-C SN when set	RW	1'b0
19	sn7_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN 7 when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
18	sn7_pcmo_prop_dis	Disables PCMO propagation for SN 7 when set	RW	1'b0
17	sn7_cmo_prop_en	Enables CMO propagation for SN 7 when set	RW	1'b0
16	sn7_128b	Data width of SN 7 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
15:9	Reserved	Reserved	RO	-
8	sn6_metadata_dis	HNF implements metadata termination flow for SN 6 when set	RW	1'b0
7	sn6_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
6	sn6_is_chie	SN 6 supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
5	sn6_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
4	sn6_sn_is_chic	Indicates that SN6 is a CHI-C SN when set	RW	1'b0
3	sn6_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN 6 when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
2	sn6_pcmo_prop_dis	Disables PCMO propagation for SN 6 when set	RW	1'b0
1	sn6_cmo_prop_en	Enables CMO propagation for SN 6 when set	RW	1'b0
0	sn6_128b	Data width of SN 6 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0

por_hnf_cml_port_aggr_grp_5-4_add_mask

This register repeats -1 times. It parametrized by the index from 5 to 4. Configures the CCIX port aggregation address mask for group 0.

Its characteristics are:

The following image shows the higher register bit assignments.

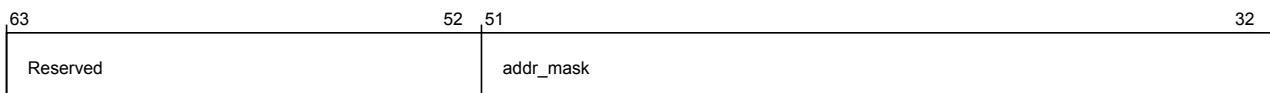


Figure 3-1316 por_hnf_por_hnf_cml_port_aggr_grp_5-4_add_mask (high)

The following table shows the por_hnf_cml_port_aggr_grp_5-4_add_mask higher register bit assignments.

Table 3-1336 por_hnf por_hnf_cml_port_aggr_grp_5-4_add_mask (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	<p>Address mask to be applied before hashing</p>	RW	46'h3FFFFFFFFF

The following image shows the lower register bit assignments.

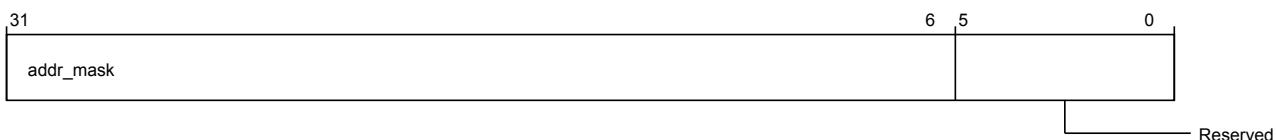


Figure 3-1317 por hnf por hnf cml port aggr qrp 5-4 add mask (low)

The following table shows the por_hnf_cml_port_aggr_grp_5-4_add_mask lower register bit assignments.

Table 3-1337 por_hnf_por_hnf_cml_port_aggr_grp_5-4_add_mask (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask	<p>Address mask to be applied before hashing</p>	RW	46'hxFFFFFFFFFFFF
5:0	Reserved	Reserved	RO	-

por_hnf_cml_port_aggr_grp_5-4_add_mask

This register repeats -1 times. It parametrized by the index from 5 to 4. Configures the CCIX port aggregation address mask for group 0.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hF80

Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the
--------------------------	---

Secure group por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.



Figure 3-1318 por hnf por hnf cml port agar grp 5-4 add mask (high)

The following table shows the por_hnf_cml_port_aggr_grp_5-4_add_mask higher register bit assignments.

Table 3-1338 por hnf por hnf cml port agar arp 5-4 add mask (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask	<p>Address mask to be applied before hashing</p>	RW	46'hxFFFFFFFFFFFF

The following image shows the lower register bit assignments.

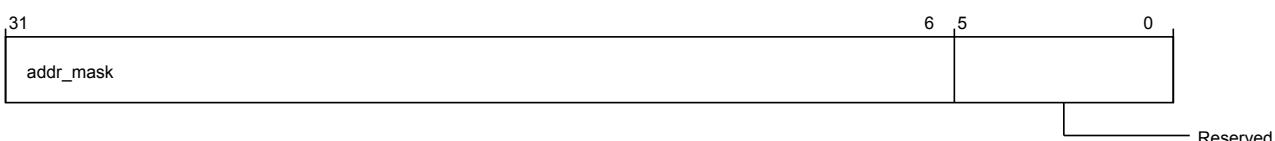


Figure 3-1319 por hnf por hnf cml port aggr qrp 5-4 add mask (low)

The following table shows the por_hnf_cml_port_aggr_grp_5-4_add_mask lower register bit assignments.

Table 3-1339 por_hnf_por_hnf_cml_port_aggr_grp_5-4_add_mask (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask	<p>Address mask to be applied before hashing</p>	RW	46'h3FFFFFFFFFFF
5:0	Reserved	Reserved	RO	-

por_hnf_cml_port_aggr_grp_5-31_add_mask

This register repeats 26 times. It parametrized by the index from 5 to 31. Configures the CCIX port aggregation address mask for group 0.

Its characteristics are:

The following image shows the higher register bit assignments.

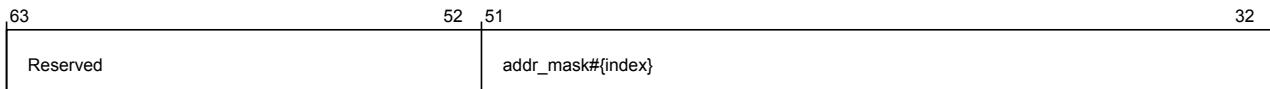


Figure 3-1320 por hnf por hnf cml port agg grp 5-31 add mask (high)

The following table shows the por_hnf_cml_port_aggr_grp_5-31_add_mask higher register bit assignments.

Table 3-1340 por hnf por hnf cml port aggr grp 5-31 add mask (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask#{index}	<p>Address mask to be applied before hashing</p>	RW	46'h3FFFFFFFFFFF

The following image shows the lower register bit assignments.

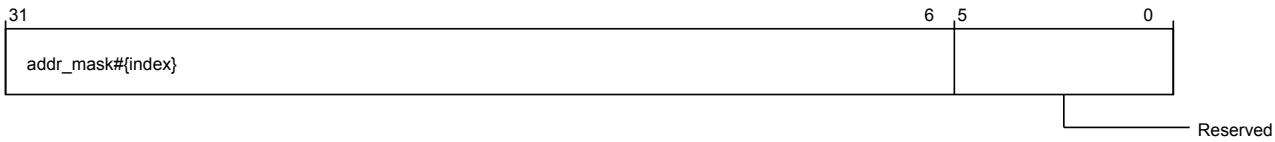


Figure 3-1321 por_hnf_por_hnf_cml_port_aggr_grp_5-31_add_mask (low)

The following table shows the por_hnf_cml_port_aggr_grp_5-31_add_mask lower register bit assignments.

Table 3-1341 por_hnf_por_hnf_cml_port_aggr_grp_5-31_add_mask (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask#{index}	<p>Address mask to be applied before hashing</p>	RW	46'h3FFFFFFFFFFF
5:0	Reserved	Reserved	RO	-

por_hnf_cml_port_aggr_grp_5-31_add_mask

This register repeats 26 times. It parametrized by the index from 5 to 31. Configures the CCIX port aggregation address mask for group 0.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'hF80

Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the
--------------------------	---

Secure group por_hnf_secure_register_groups_override.sam_control

override

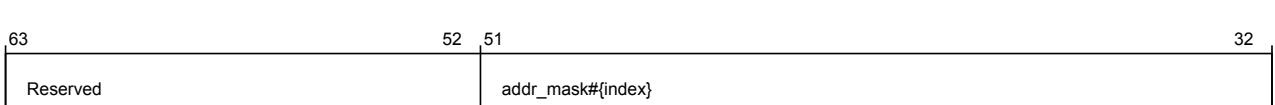


Figure 3-1322 por hnf por hnf cml port aggr grp 5-31 add mask (high)

The following table shows the por_hnf_cml_port_aggr_grp_5-31_add_mask higher register bit assignments.

Table 3-1342 por hnf por hnf cml port aggr grp 5-31 add mask (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	addr_mask#{index}	<p>Address mask to be applied before hashing</p>	RW	46'h3FFFFFFFFFFF

The following image shows the lower register bit assignments.

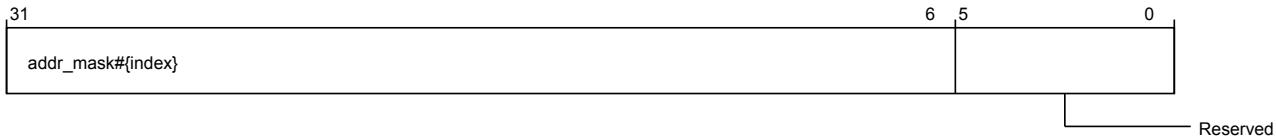


Figure 3-1323 por_hnf_por_hnf_cml_port_aggr_grp_5-31_add_mask (low)

The following table shows the por_hnf_cml_port_aggr_grp_5-31_add_mask lower register bit assignments.

Table 3-1343 por_hnf_por_hnf_cml_port_aggr_grp_5-31_add_mask (low)

Bits	Field name	Description	Type	Reset
31:6	addr_mask#{index}	<p>Address mask to be applied before hashing</p>	RW	46'h3FFFFFFFFF
5:0	Reserved	Reserved	RO	-

por_hnf_cml_port_aggr_grp_reg_2-12

This register repeats 10 times. It parametrized by the index from 2 to 12. Configures the CCIX port aggregation port Node IDs.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hFB0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

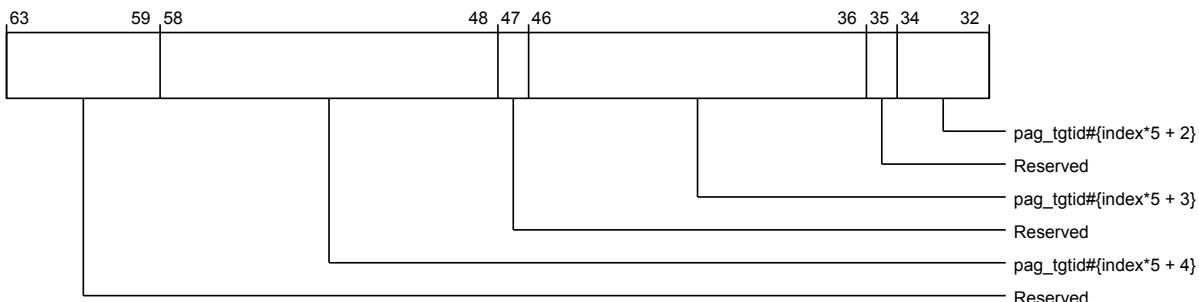


Figure 3-1324 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (high)

The following table shows the por_hnf_cml_port_aggr_grp_reg_2-12 higher register bit assignments.

Table 3-1344 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	pag_tgtid#{index*5 + 4}	<p>Specifies the target ID #{index*5 + 4} for CPAG</p>	RW	11'b0
47	Reserved	Reserved	RO	-
46:36	pag_tgtid#{index*5 + 3}	<p>Specifies the target ID #{index*5 + 3} for CPAG</p>	RW	11'b0
35	Reserved	Reserved	RO	-
34:32	pag_tgtid#{index*5 + 2}	<p>Specifies the target ID {index*5 + 2} for CPAG</p>	RW	11'b0

The following image shows the lower register bit assignments.

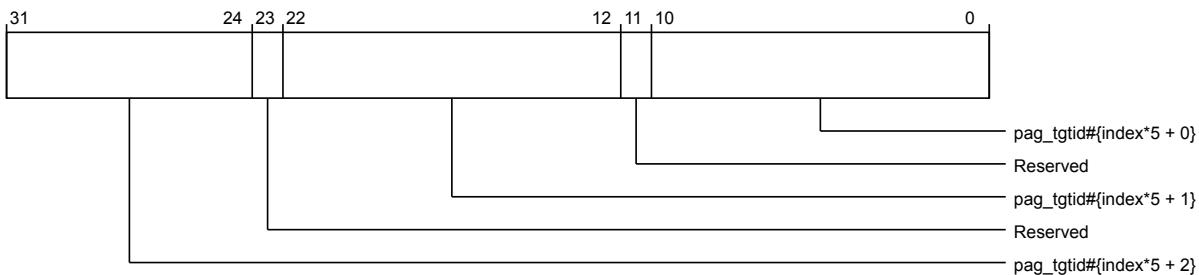


Figure 3-1325 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (low)

The following table shows the por_hnf_cml_port_aggr_grp_reg_2-12 lower register bit assignments.

Table 3-1345 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (low)

Bits	Field name	Description	Type	Reset
31:24	pag_tgtid#{index*5 + 2}	<p>Specifies the target ID {index*5 + 2} for CPAG</p>	RW	11'b0
23	Reserved	Reserved	RO	-
22:12	pag_tgtid#{index*5 + 1}	<p>Specifies the target ID {index*5 + 1} for CPAG</p>	RW	11'b0
11	Reserved	Reserved	RO	-
10:0	pag_tgtid#{index*5 + 0}	<p>Specifies the target ID {index*5 + 0} for CPAG</p>	RW	11'b0

por_hnf_cml_port_aggr_grp_reg_2-12

This register repeats 10 times. It parametrized by the index from 2 to 12. Configures the CCIX port aggregation port Node IDs.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hFB0
Register reset	64'b0

Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

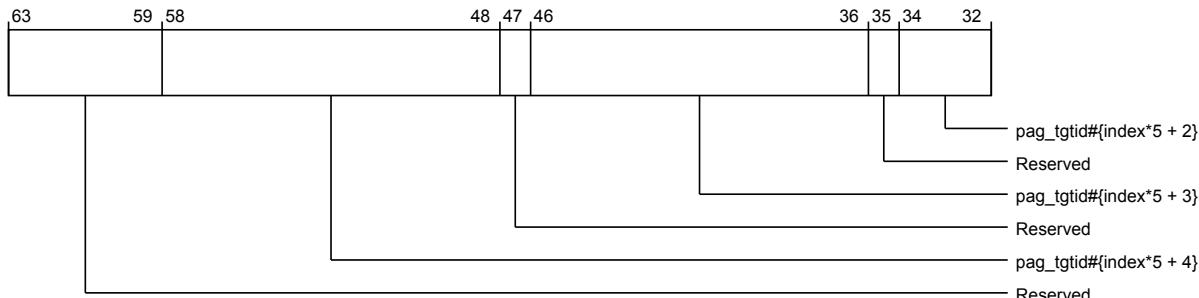


Figure 3-1326 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (high)

The following table shows the por_hnf_cml_port_aggr_grp_reg_2-12 higher register bit assignments.

Table 3-1346 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	pag_tgtid#{index*5 + 4}	<p>Specifies the target ID #{index*5 + 4} for CPAG</p>	RW	11'b0
47	Reserved	Reserved	RO	-
46:36	pag_tgtid#{index*5 + 3}	<p>Specifies the target ID #{index*5 + 3} for CPAG</p>	RW	11'b0
35	Reserved	Reserved	RO	-
34:32	pag_tgtid#{index*5 + 2}	<p>Specifies the target ID {index*5 + 2} for CPAG</p>	RW	11'b0

The following image shows the lower register bit assignments.

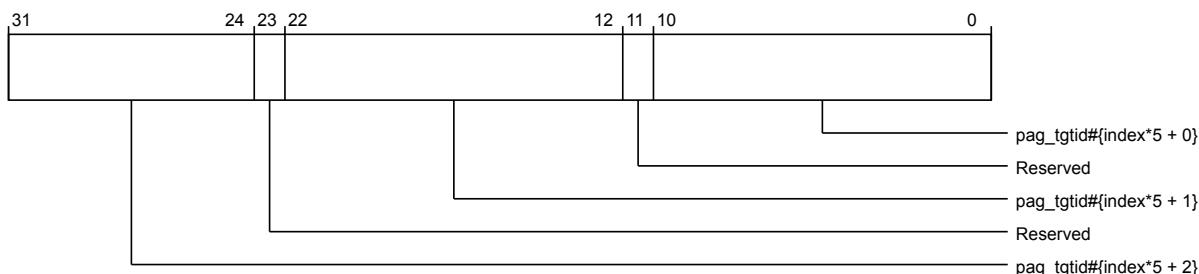


Figure 3-1327 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (low)

The following table shows the por_hnf_cml_port_aggr_grp_reg_2-12 lower register bit assignments.

Table 3-1347 por_hnf_por_hnf_cml_port_aggr_grp_reg_2-12 (low)

Bits	Field name	Description	Type	Reset
31:24	pag_tgtid#{index*5 + 2}	<p>Specifies the target ID {index*5 + 2} for CPAG</p>	RW	11'b0
23	Reserved	Reserved	RO	-
22:12	pag_tgtid#{index*5 + 1}	<p>Specifies the target ID {index*5 + 1} for CPAG</p>	RW	11'b0
11	Reserved	Reserved	RO	-
10:0	pag_tgtid#{index*5 + 0}	<p>Specifies the target ID {index*5 + 0} for CPAG</p>	RW	11'b0

por_hnf_cml_port_aggr_ctrl_reg

Configures the CCIX port aggregation port groups

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hFD0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

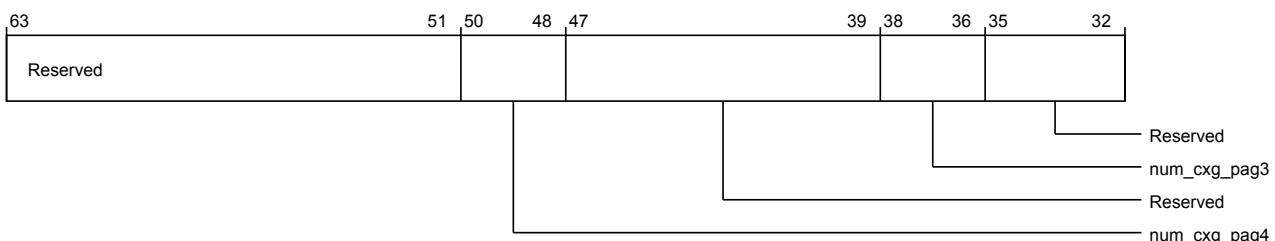


Figure 3-1328 por_hnf_por_hnf_cml_port_aggr_ctrl_reg (high)

The following table shows the por_hnf_cml_port_aggr_ctrl_reg higher register bit assignments.

Table 3-1348 por_hnf_por_hnf_cml_port_aggr_ctrl_reg (high)

Bits	Field name	Description	Type	Reset
63:51	Reserved	Reserved	RO	-
50:48	num_cxg_pag4	Specifies the number of CXRAs in CPAG4 Constraint: May use pag_tgtid8 through pag_tgtid9 of por_hnf_cml_port_aggr_grp_reg1 when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0 3'b000: 1 port used 3'b001: 2 ports used 3'b010: 4 ports used 3'b011: 8 ports used 3'b100: 16 ports used 3'b101: 32 ports used 3'b110: Reserved 3'b111: Reserved	RW	3'b000
47:39	Reserved	Reserved	RO	-
38:36	num_cxg_pag3	Specifies the number of CXRAs in CPAG3 Constraint: May use pag_tgtid6 through pag_tgtid7 of por_hnf_cml_port_aggr_grp_reg1 when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0 3'b000: 1 port used 3'b001: 2 ports used 3'b010: 4 ports used 3'b011: 8 ports used 3'b100: 16 ports used 3'b101: 32 ports used 3'b110: Reserved 3'b111: Reserved	RW	3'b000
35:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

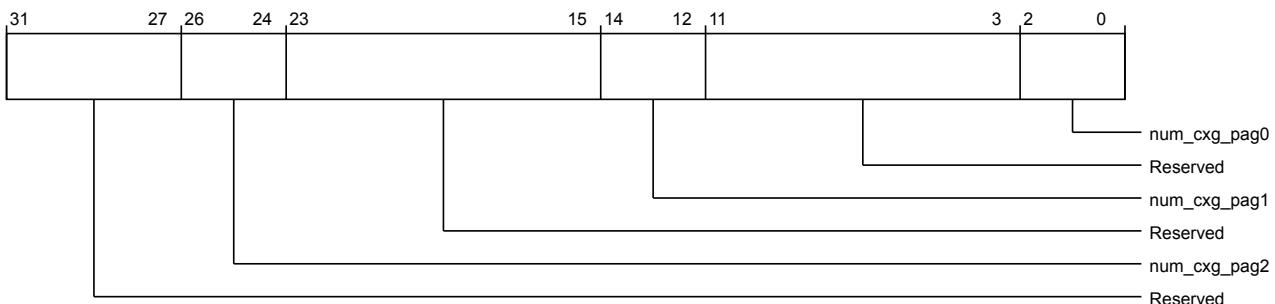


Figure 3-1329 por_hnf_por_hnf_cml_port_aggr_ctrl_reg (low)

The following table shows the por_hnf_cml_port_aggr_ctrl_reg lower register bit assignments.

Table 3-1349 por_hnf_por_hnf_cml_port_aggr_ctrl_reg (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:24	num_cxg_pag2	<p>Specifies the number of CXRAs in CPAG2</p> <p>Constraint: May use pag_tgtid4 through pag_tgtid7 of por_hnf_cml_port_aggr_grp_reg[0,1] when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0</p> <p>3'b000: 1 port used 3'b001: 2 ports used 3'b010: 4 ports used 3'b011: 8 ports used 3'b100: 16 ports used 3'b101: 32 ports used 3'b110: Reserved 3'b111: Reserved</p>	RW	3'b000
23:15	Reserved	Reserved	RO	-
14:12	num_cxg_pag1	<p>Specifies the number of CXRAs in CPAG1</p> <p>Constraint: May use pag_tgtid2 through pag_tgtid3 of por_hnf_cml_port_aggr_grp_reg0 when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0</p> <p>3'b000: 1 port used 3'b001: 2 ports used 3'b010: 4 ports used 3'b011: 8 ports used 3'b100: 16 ports used 3'b101: 32 ports used 3'b110: Reserved 3'b111: Reserved</p>	RW	3'b000

Table 3-1349 por_hnf_por_hnf_cml_port_aggr_ctrl_reg (low) (continued)

Bits	Field name	Description	Type	Reset
11:3	Reserved	Reserved	RO	-
2:0	num_cxg_pag0	<p>Specifies the number of CXRAs in CPAG0</p> <p>Constraint: May use pag_tgtid0 through pag_tgtid7 of por_hnf_cml_port_aggr_grp_reg[0,1] when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0</p> <ul style="list-style-type: none"> 3'b000: 1 port used 3'b001: 2 ports used 3'b010: 4 ports used 3'b011: 8 ports used 3'b100: 16 ports used 3'b101: 32 ports used 3'b110: Reserved 3'b111: Reserved 	RW	3'b000

por_hnf_cml_port_aggr_ctrl_reg_1-6

This register repeats 5 times. It parametrized by the index from 1 to 6. Configures the CCIX port aggregation port groups

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h6200 + # {8 * [1, 2, 3, 4, 5, 6]}

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

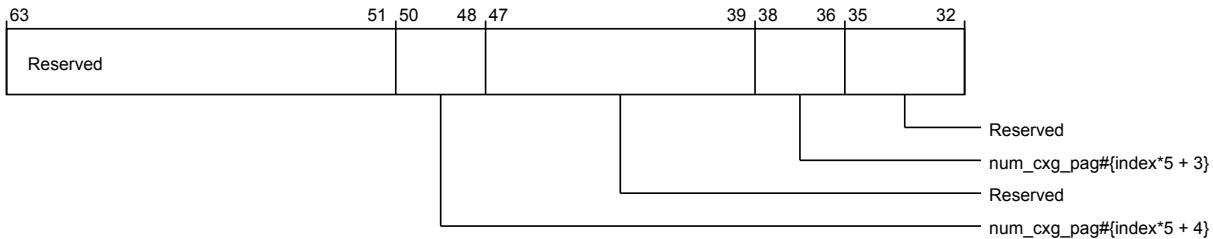


Figure 3-1330 por_hnf_por_hnf_cml_port_aggr_ctrl_reg_1-6 (high)

The following table shows the port assignments for the higher register bit assignments.

Table 3-1350 por_hnf_por_hnf_cml_port_aggr_ctrl_reg_1-6 (high)

Bits	Field name	Description	Type	Reset
63:51	Reserved	Reserved	RO	-
50:48	num_cxg_pag#{index*5 + 4}	<p>Specifies the number of CXRAs in CPAG4#{index*5 + 4}</p> <p>Constraint: May use pag_tgtid8 through pag_tgtid9 of por_hnf_cml_port_aggr_grp_reg1 when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0</p> <p>3'b000: 1 port used</p> <p>3'b001: 2 ports used</p> <p>3'b010: 4 ports used</p> <p>3'b011: 8 ports used</p> <p>3'b100: 16 ports used</p> <p>3'b101: 32 ports used</p> <p>3'b110: Reserved</p> <p>3'b111: Reserved</p>	RW	3'b000
47:39	Reserved	Reserved	RO	-
38:36	num_cxg_pag#{index*5 + 3}	<p>Specifies the number of CXRAs in CPAG3#{index*5 + 3}</p> <p>Constraint: May use pag_tgtid6 through pag_tgtid7 of por_hnf_cml_port_aggr_grp_reg1 when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0</p> <p>3'b000: 1 port used</p> <p>3'b001: 2 ports used</p> <p>3'b010: 4 ports used</p> <p>3'b011: 8 ports used</p> <p>3'b100: 16 ports used</p> <p>3'b101: 32 ports used</p> <p>3'b110: Reserved</p> <p>3'b111: Reserved</p>	RW	3'b000
35:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

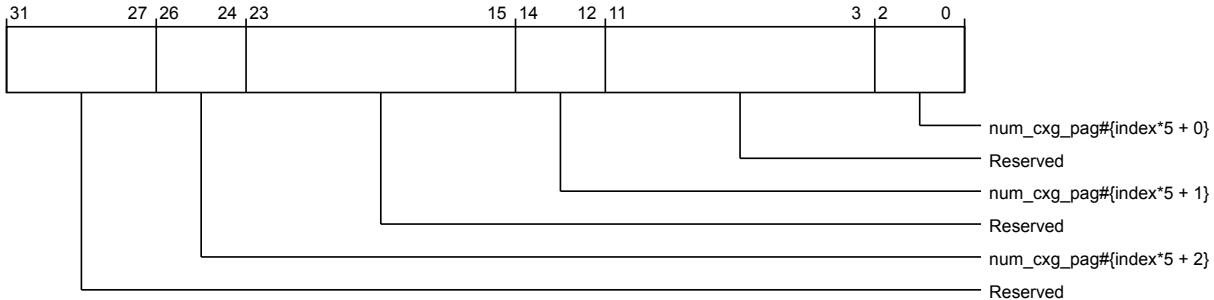


Figure 3-1331 por_hnf_por_hnf_cml_port_aggr_ctrl_reg_1-6 (low)

The following table shows the por_hnf_cml_port_aggr_ctrl_reg_1-6 lower register bit assignments.

Table 3-1351 por_hnf_por_hnf_cml_port_aggr_ctrl_reg_1-6 (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:24	num_cxg_pag#{index*5 + 2}	<p>Specifies the number of CXRAs in CPAG2#{index*5 + 2}</p> <p>Constraint: May use pag_tgtid4 through pag_tgtid7 of por_hnf_cml_port_aggr_grp_reg[0,1] when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0</p> <p>3'b000: 1 port used</p> <p>3'b001: 2 ports used</p> <p>3'b010: 4 ports used</p> <p>3'b011: 8 ports used</p> <p>3'b100: 16 ports used</p> <p>3'b101: 32 ports used</p> <p>3'b110: Reserved</p> <p>3'b111: Reserved</p>	RW	3'b000
23:15	Reserved	Reserved	RO	-
14:12	num_cxg_pag#{index*5 + 1}	<p>Specifies the number of CXRAs in CPAG1#{index*5 + 1}</p> <p>Constraint: May use pag_tgtid2 through pag_tgtid3 of por_hnf_cml_port_aggr_grp_reg0 when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0</p> <p>3'b000: 1 port used</p> <p>3'b001: 2 ports used</p> <p>3'b010: 4 ports used</p> <p>3'b011: 8 ports used</p> <p>3'b100: 16 ports used</p> <p>3'b101: 32 ports used</p> <p>3'b110: Reserved</p> <p>3'b111: Reserved</p>	RW	3'b000

Table 3-1351 por_hnf_por_hnf_cml_port_aggr_ctrl_reg_1-6 (low) (continued)

Bits	Field name	Description	Type	Reset
11:3	Reserved	Reserved	RO	-
2:0	num_cxg_pag#{index*5 + 0}	Specifies the number of CXRAs in CPAG#{index*5 + 0} Constraint: May use pag_tgtid0 through pag_tgtid7 of por_hnf_cml_port_aggr_grp_reg[0,1] when POR_RNSAM_FLEX_TGTID_EN_PARAM = 0 3'b000: 1 port used 3'b001: 2 ports used 3'b010: 4 ports used 3'b011: 8 ports used 3'b100: 16 ports used 3'b101: 32 ports used 3'b110: Reserved 3'b111: Reserved	RW	3'b000

por_hnf_abf_lo_addr

Lower address range for Address Based Flush (ABF) [51:0].

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF50
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.ppu

The following image shows the higher register bit assignments.

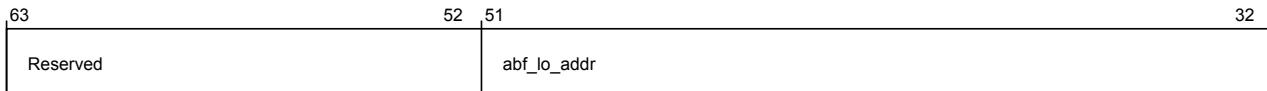


Figure 3-1332 por_hnf_por_hnf_abf_lo_addr (high)

The following table shows the por_hnf_abf_lo_addr higher register bit assignments.

Table 3-1352 por_hnf_por_hnf_abf_lo_addr (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	abf_lo_addr	Lower address range for ABF	RW	52'b0

The following image shows the lower register bit assignments.



Figure 3-1333 por_hnf_por_hnf_abf_lo_addr (low)

The following table shows the por_hnf_abf_lo_addr lower register bit assignments.

Table 3-1353 por_hnf_por_hnf_abf_lo_addr (low)

Bits	Field name	Description	Type	Reset
31:0	abf_lo_addr	Lower address range for ABF	RW	52'b0

por_hnf_abf_hi_addr

Upper address range for Address Based Flush (ABF) [51:0].

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF58
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.ppu

The following image shows the higher register bit assignments.

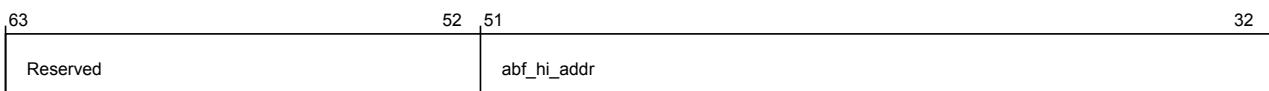


Figure 3-1334 por_hnf_por_hnf_abf_hi_addr (high)

The following table shows the por_hnf_abf_hi_addr higher register bit assignments.

Table 3-1354 por_hnf_por_hnf_abf_hi_addr (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	abf_hi_addr	Upper address range for ABF	RW	52'b0

The following image shows the lower register bit assignments.

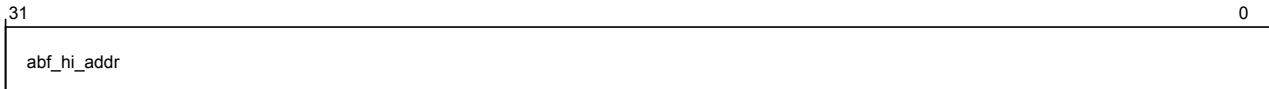


Figure 3-1335 por_hnf_por_hnf_abf_hi_addr (low)

The following table shows the por_hnf_abf_hi_addr lower register bit assignments.

Table 3-1355 por_hnf_por_hnf_abf_hi_addr (low)

Bits	Field name	Description	Type	Reset
31:0	abf_hi_addr	Upper address range for ABF	RW	52'b0

por_hnf_abf_pr

Functions as the Address Based Flush (ABF) policy register.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hF60
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.ppu

The following image shows the higher register bit assignments.

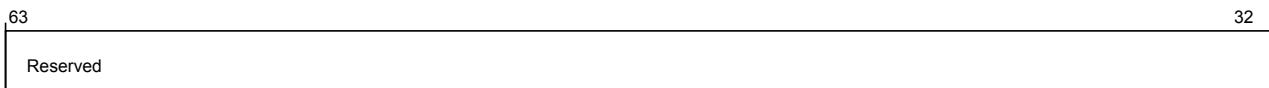


Figure 3-1336 por_hnf_por_hnf_abf_pr (high)

The following table shows the por_hnf_abf_pr higher register bit assignments.

Table 3-1356 por_hnf_por_hnf_abf_pr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

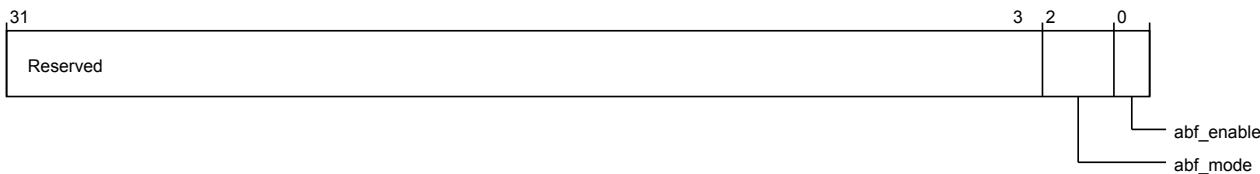


Figure 3-1337 por_hnf_por_hnf_abf_pr (low)

The following table shows the por_hnf_abf_pr lower register bit assignments.

Table 3-1357 por_hnf_por_hnf_abf_pr (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2:1	abf_mode	ABF mode 2'b00: Clean Invalidate; WB dirty data and invalidate local copy 2'b01: Make Invalidate; invalidate without writing back dirty data 2'b10: Clean Shared; WB dirty data and can keep clean copy 2'b11: Reserved	RW	2'b00
0	abf_enable	Start Address Based Flushing based on high and low address ranges	RW	1'b0

por_hnf_abf_sr

Functions as the Address Based Flush (ABF) status register.

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hF68

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

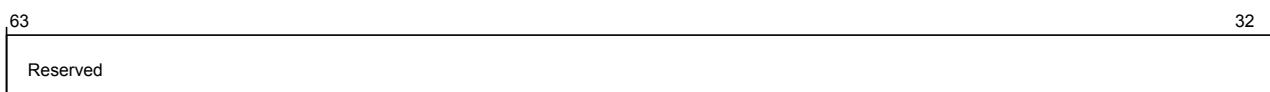


Figure 3-1338 por_hnf_por_hnf_abf_sr (high)

The following table shows the por_hnf_abf_sr higher register bit assignments.

Table 3-1358 por_hnf_por_hnf_abf_sr (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

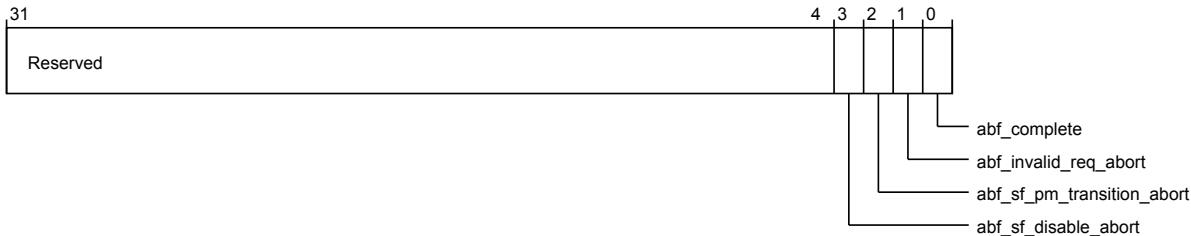


Figure 3-1339 por_hnf_por_hnf_abf_sr (low)

The following table shows the por_hnf_abf_sr lower register bit assignments.

Table 3-1359 por_hnf_por_hnf_abf_sr (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3	abf_sf_disable_abort	ABF aborted due to SF not being enabled, either by configuration or double-bit ECC error	RO	1'b0
2	abf_sf_pm_transition_abort	ABF aborted due to PM transition while ABF in progress, or both PM and ABF requested at the same time	RO	1'b0
1	abf_invalid_req_abort	ABF request made while PM is not in FAM/HAM/SF_ONLY mode; request aborted in this case	RO	1'b0
0	abf_complete	ABF completed	RO	1'b0

por_hnf_cbusy_write_limit_ctl

Cbusy threshold limits for POCQ write entries. CONSTRAINT: The hnf_adv_cbusy_mode_dis must be 1'b0 to use this feature.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1000
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses. This register can be modified only with prior written permission from Arm.

The following image shows the higher register bit assignments.

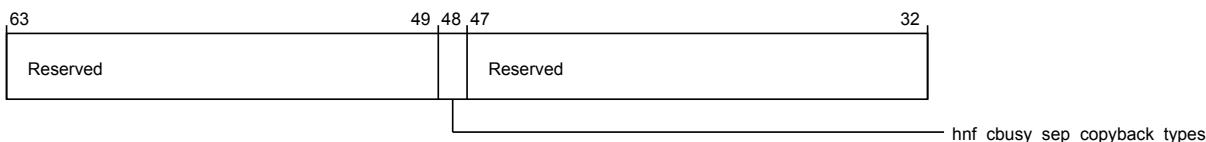


Figure 3-1340 por_hnf_por_hnf_cbusy_write_limit_ctl (high)

The following table shows the por_hnf_cbusy_write_limit_ctl higher register bit assignments.

Table 3-1360 por_hnf_por_hnf_cbusy_write_limit_ctl (high)

Bits	Field name	Description	Type	Reset
63:49	Reserved	Reserved	RO	-
48	hnf_cbusy_sep_copyback_types	Enables copyback and non-copyback write type separation in cbusy calculation	RW	1'b0
47:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

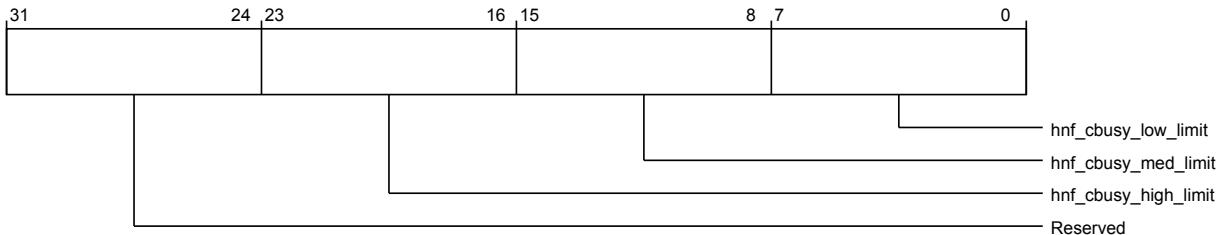


Figure 3-1341 por_hnf_por_hnf_cbusy_write_limit_ctl (low)

The following table shows the por_hnf_cbusy_write_limit_ctl lower register bit assignments.

Table 3-1361 por_hnf_por_hnf_cbusy_write_limit_ctl (low)

Bits	Field name	Description	Type	Reset
31:24	Reserved	Reserved	RO	-
23:16	hnf_cbusy_high_limit	POCQ limit for Write CBusy High	RW	Configuration dependent
15:8	hnf_cbusy_med_limit	POCQ limit for Write CBusy Med	RW	Configuration dependent
7:0	hnf_cbusy_low_limit	POCQ limit for Write CBusy Low	RW	Configuration dependent

por_hnf_cbusy_resp_ctl

Controls the responses sent from HNF to RNF. CONSTRAINT: The hnf_adv_cbusy_mode_dis must be 1'b0 to use this feature.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1008
Register reset	64'b0000100000
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

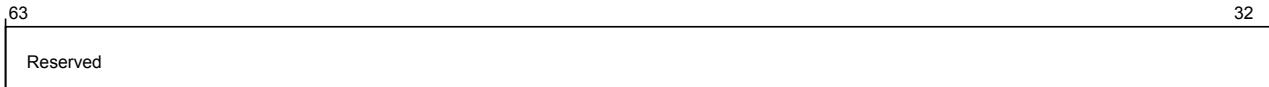


Figure 3-1342 por_hnf_por_hnf_cbusy_resp_ctl (high)

The following table shows the por_hnf_cbusy_resp_ctl higher register bit assignments.

Table 3-1362 por_hnf_por_hnf_cbusy_resp_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

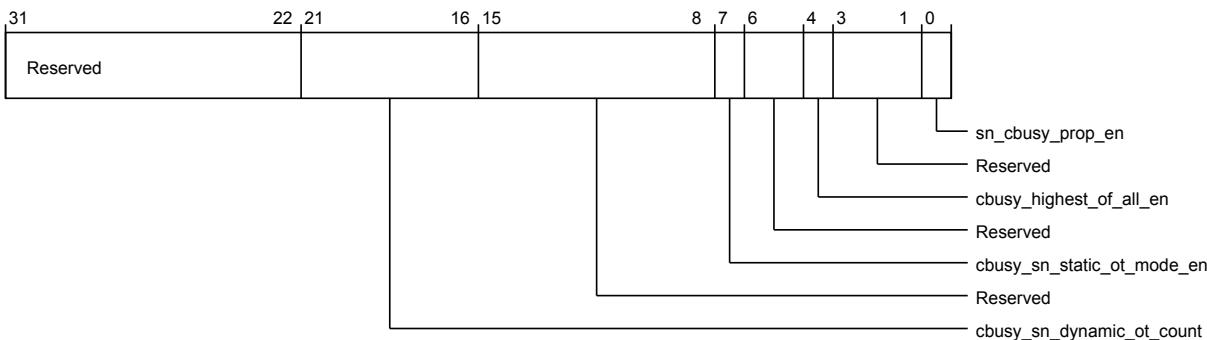


Figure 3-1343 por_hnf_por_hnf_cbusy_resp_ctl (low)

The following table shows the por_hnf_cbusy_resp_ctl lower register bit assignments.

Table 3-1363 por_hnf_por_hnf_cbusy_resp_ctl (low)

Bits	Field name	Description	Type	Reset
31:22	Reserved	Reserved	RO	-
21:16	cbusy_sn_dynamic_ot_count	Specifies the granularity at which HN-F will dynamically throttle transactions to SN-F. CONSTRAINT: 2,4,8 are the allowed values	RW	6'b000100
15:8	Reserved	Reserved	RO	-
7	cbusy_sn_static_ot_mode_en	Controls cbusy between HN-F and SN-F 1'b0: HN-F will dynamically throttle outstanding requests to SN-F 1'b1: HN-F will use fixed transactions count at each CBusy level at 1/4th POCQ granularity	RW	1'b0
6:5	Reserved	Reserved	RO	-

Table 3-1363 por_hnf_por_hnf_cbusy_resp_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
4	cbusy_highest_of_all_en	Controls cbusy between HN-F and SN-F 1'b0: Will send the HN-F or SN-F as configured 1'b1: Will select highest CBusy value between the SN-F and HN-F	RW	1'b0
3:1	Reserved	Reserved	RO	-
0	sn_cbusy_prop_en	Controls HN-F and SN-F cbusy on responses to RN-F 1'b0: HN-F's POCQ Cbusy is sent 1'b1: SN-F's Cbusy is sent	RW	1'b0

por_hnf_cbusy_sn_ctl

Controls the SN-F cbusy thresholds. CONSTRAINT: The hnf_adv_cbusy_mode_dis must be 1'b0 to use this feature.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1010

Register reset 64'b0100000000000000100000000100000000100000000

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

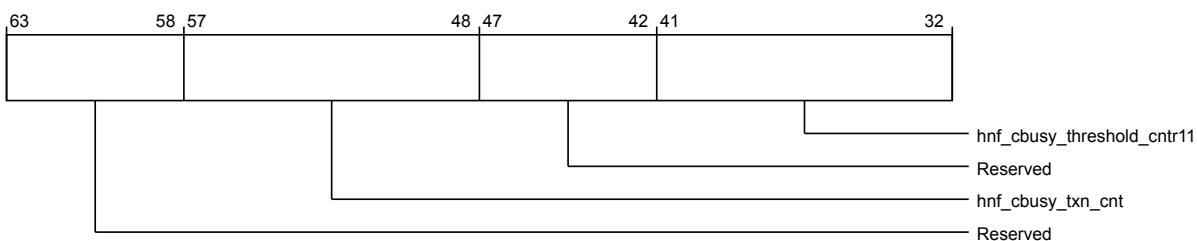


Figure 3-1344 por_hnf_por_hnf_cbusy_sn_ctl (high)

The following table shows the por_hnf_cbusy_sn_ctl higher register bit assignments.

Table 3-1364 por_hnf_por_hnf_cbusy_sn_ctl (high)

Bits	Field name	Description	Type	Reset
63:58	Reserved	Reserved	RO	-
57:48	hnf_cbusy_txn_cnt	Number of transactions over which the counters are tracked	RW	10'b0100000000
47:42	Reserved	Reserved	RO	-
41:32	hnf_cbusy_threshold_cntr11	CBusy threshold at which SN-F is considered busy for Counter_11	RW	10'b0000010000

The following image shows the lower register bit assignments.

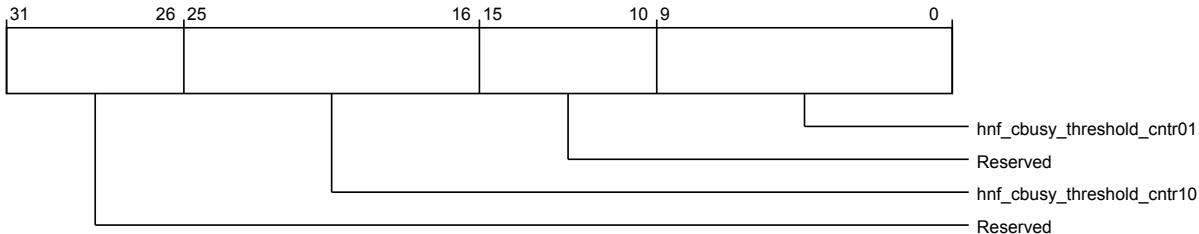


Figure 3-1345 por_hnf_por_hnf_cbusy_sn_ctl (low)

The following table shows the por_hnf_cbusy_sn_ctl lower register bit assignments.

Table 3-1365 por_hnf_por_hnf_cbusy_sn_ctl (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:16	hnf_cbusy_threshold_cntr10	CBusy threshold at which SN-F is considered busy for Counter_10	RW	10'b0000100000
15:10	Reserved	Reserved	RO	-
9:0	hnf_cbusy_threshold_cntr01	CBusy threshold at which SN-F is considered busy for Counter_01	RW	10'b0001000000

por_hnf_pocq_alloc_class_dedicated

Controls Dedicated entries in POCQ for each class.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1020

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

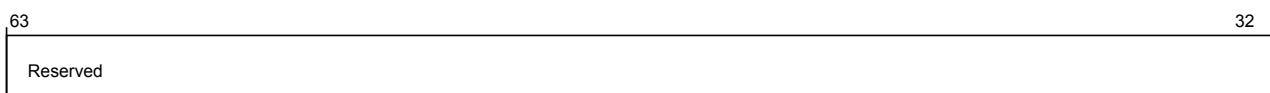


Figure 3-1346 por_hnf_por_hnf_pocq_alloc_class_dedicated (high)

The following table shows the por_hnf_pocq_alloc_class_dedicated higher register bit assignments.

Table 3-1366 por_hnf_por_hnf_pocq_alloc_class_dedicated (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

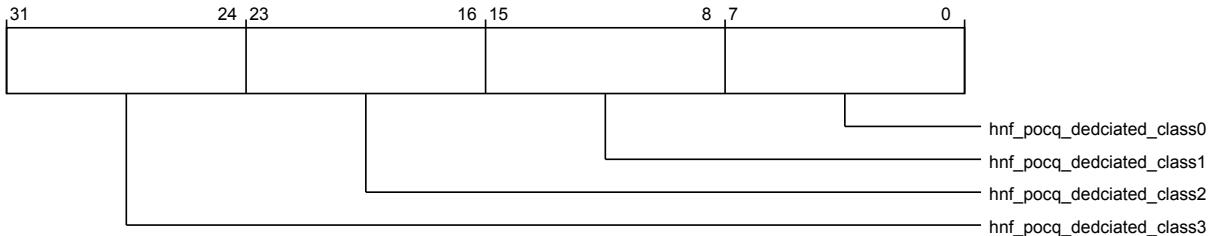


Figure 3-1347 por_hnf_por_hnf_pocq_alloc_class_dedicated (low)

The following table shows the por_hnf_pocq_alloc_class_dedicated lower register bit assignments.

Table 3-1367 por_hnf_por_hnf_pocq_alloc_class_dedicated (low)

Bits	Field name	Description	Type	Reset
31:24	hnf_pocq_dedicated_class3	Dedicated number of entries for Class 3 in POCQ CONSTRAINT: Sum of dedicated entries for classes and SEQ can not exceed HNF_NUM_ENTRIES_POCQ_PARAM. CONSTRAINT: hnf_pocq_dedicated_class3 < hnf_pocq_max_allowed_class3	RW	8'b00000000
23:16	hnf_pocq_dedicated_class2	Dedicated number of entries for Class 2 in POCQ CONSTRAINT: Sum of dedicated entries for classes and SEQ can not exceed HNF_NUM_ENTRIES_POCQ_PARAM. CONSTRAINT: hnf_pocq_dedicated_class2 < hnf_pocq_max_allowed_class2	RW	8'b00000000
15:8	hnf_pocq_dedicated_class1	Dedicated number of entries for Class 1 in POCQ CONSTRAINT: Sum of dedicated entries for classes and SEQ can not exceed HNF_NUM_ENTRIES_POCQ_PARAM. CONSTRAINT: hnf_pocq_dedicated_class1 < hnf_pocq_max_allowed_class1	RW	8'b00000000
7:0	hnf_pocq_dedicated_class0	Dedicated number of entries for Class 0 in POCQ CONSTRAINT: Sum of dedicated entries for classes and SEQ can not exceed HNF_NUM_ENTRIES_POCQ_PARAM. CONSTRAINT: hnf_pocq_dedicated_class0 < hnf_pocq_max_allowed_class0	RW	8'b00000000

por_hnf_pocq_alloc_class_max_allowed

Controls Maximum allowed entries in POCQ for each class.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1028
Register reset	Configuration dependent

Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.



Figure 3-1348 por_hnf_por_hnf_pocq_alloc_class_max_allowed (high)

The following table shows the por_hnf_pocq_alloc_class_max_allowed higher register bit assignments.

Table 3-1368 por_hnf_por_hnf_pocq_alloc_class_max_allowed (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

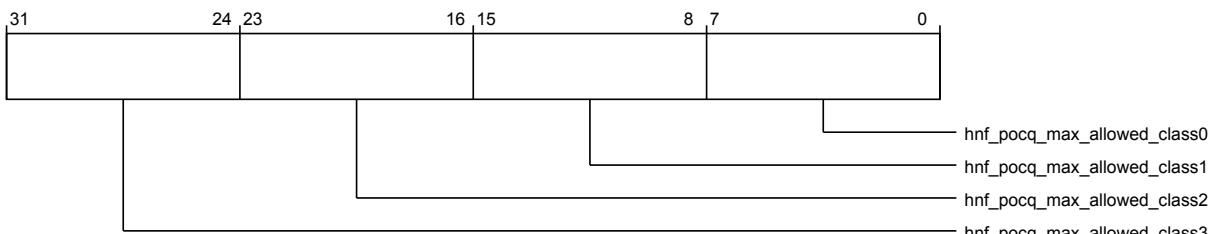


Figure 3-1349 por_hnf_por_hnf_pocq_alloc_class_max_allowed (low)

The following table shows the por_hnf_pocq_alloc_class_max_allowed lower register bit assignments.

Table 3-1369 por_hnf_por_hnf_pocq_alloc_class_max_allowed (low)

Bits	Field name	Description	Type	Reset
31:24	hnf_pocq_max_allowed_class3	Maximum number of entries for Class 3 in POCQ CONSTRAINT: hnf_pocq_dedicated_class3 < hnf_pocq_max_allowed_class3	RW	Configuration dependent
23:16	hnf_pocq_max_allowed_class2	Maximum number of entries for Class 2 in POCQ CONSTRAINT: hnf_pocq_dedicated_class2 < hnf_pocq_max_allowed_class2	RW	Configuration dependent
15:8	hnf_pocq_max_allowed_class1	Maximum number of entries for Class 1 in POCQ CONSTRAINT: hnf_pocq_dedicated_class1 < hnf_pocq_max_allowed_class1	RW	Configuration dependent
7:0	hnf_pocq_max_allowed_class0	Maximum number of entries for Class 0 in POCQ CONSTRAINT: hnf_pocq_dedicated_class0 < hnf_pocq_max_allowed_class0	RW	Configuration dependent

por_hnf_pocq_alloc_class_contended_min

Controls Contended minimum entries in POCQ for each class.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1030
Register reset	Configuration dependent
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

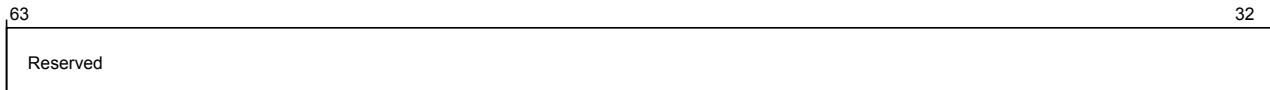


Figure 3-1350 por_hnf_pocq_alloc_class_contended_min (high)

The following table shows the por_hnf_pocq_alloc_class_contended_min higher register bit assignments.

Table 3-1370 por_hnf_pocq_alloc_class_contended_min (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

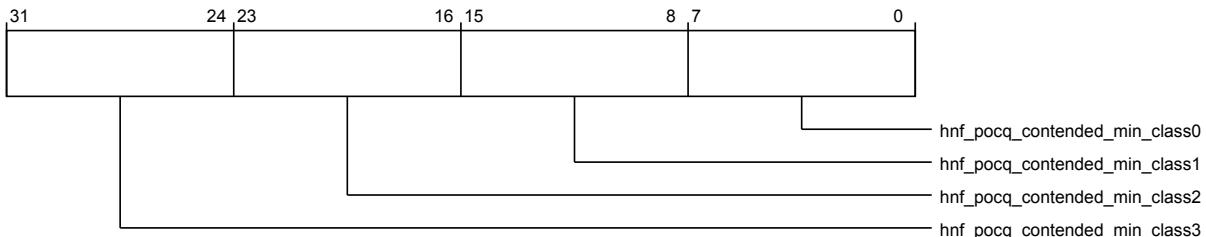


Figure 3-1351 por_hnf_pocq_alloc_class_contended_min (low)

The following table shows the por_hnf_pocq_alloc_class_contended_min lower register bit assignments.

Table 3-1371 por_hnf_pocq_alloc_class_contended_min (low)

Bits	Field name	Description	Type	Reset
31:24	hnf_pocq_contended_min_class3	Contended min entries for Class 3 in POCQ	RW	Configuration dependent
23:16	hnf_pocq_contended_min_class2	Contended min entries for Class 2 in POCQ	RW	Configuration dependent

Table 3-1371 por_hnf_por_hnf_pocq_alloc_class_contented_min (low) (continued)

Bits	Field name	Description	Type	Reset
15:8	hnf_pocq_contented_min_class1	Contented min entries for Class 1 in POCQ	RW	Configuration dependent
7:0	hnf_pocq_contented_min_class0	Contented min entries for Class 0 in POCQ	RW	Configuration dependent

por_hnf_class_ctl

Class misc controls.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1038

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

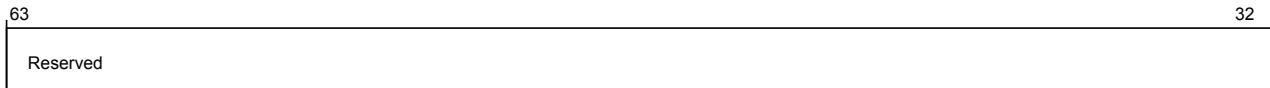


Figure 3-1352 por_hnf_por_hnf_class_ctl (high)

The following table shows the por_hnf_class_ctl higher register bit assignments.

Table 3-1372 por_hnf_por_hnf_class_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

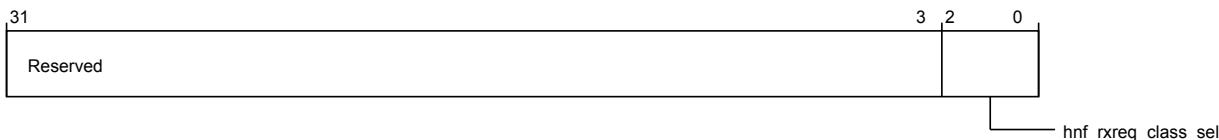


Figure 3-1353 por_hnf_por_hnf_class_ctl (low)

The following table shows the por_hnf_class_ctl lower register bit assignments.

Table 3-1373 por_hnf_por_hnf_class_ctl (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2:0	hnf_rxreq_class_sel	RxReq Class select: 3'b000: QoS based class selection 3'b001: Request Opcode based class selection Note: If un-supported value is programmed, default selection of QoS based is chosen.	RW	3'b000

por_hnf_pocq_qos_class_ctl

QoS bases class identification controls.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1040
Register reset	64'b0111000010111000111011001111111
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

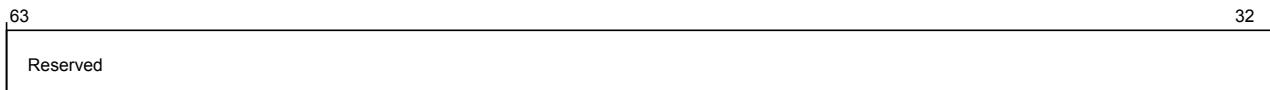


Figure 3-1354 por_hnf_por_hnf_pocq_qos_class_ctl (high)

The following table shows the por_hnf_pocq_qos_class_ctl higher register bit assignments.

Table 3-1374 por_hnf_por_hnf_pocq_qos_class_ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

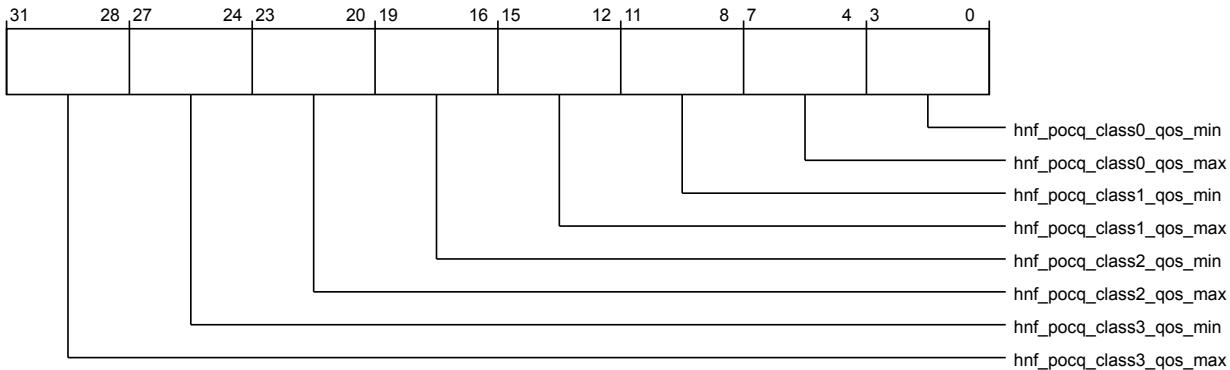


Figure 3-1355 por_hnf_por_hnf_pocq_qos_class_ctl (low)

The following table shows the por_hnf_pocq_qos_class_ctl lower register bit assignments.

Table 3-1375 por_hnf_por_hnf_pocq_qos_class_ctl (low)

Bits	Field name	Description	Type	Reset
31:28	hnf_pocq_class3_qos_max	QoS maximum value for Class 3	RW	4'b0111
27:24	hnf_pocq_class3_qos_min	QoS minimum value for Class 3	RW	4'b0000
23:20	hnf_pocq_class2_qos_max	QoS maximum value for Class 2	RW	4'b1011
19:16	hnf_pocq_class2_qos_min	QoS minimum value for Class 2	RW	4'b1000
15:12	hnf_pocq_class1_qos_max	QoS maximum value for Class 1	RW	4'b1110
11:8	hnf_pocq_class1_qos_min	QoS minimum value for Class 1	RW	4'b1100
7:4	hnf_pocq_class0_qos_max	QoS maximum value for Class 0	RW	4'b1111
3:0	hnf_pocq_class0_qos_min	QoS minimum value for Class 0	RW	4'b1111

por_hnf_class_pocq_arb_weight_ctl

Per Class weight controls for scheduling requests from POCQ.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1050
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

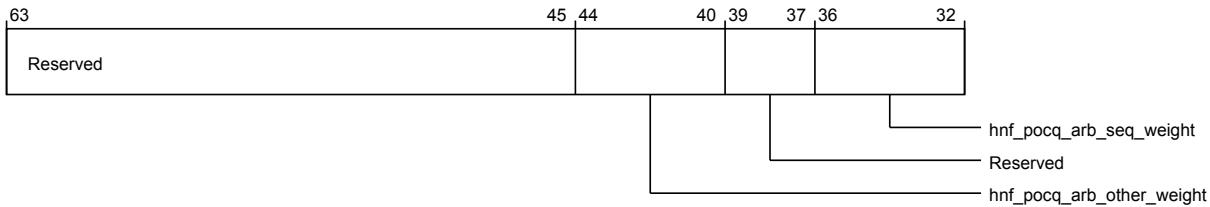


Figure 3-1356 por_hnf_por_hnf_class_pocq_arb_weight_ctl (high)

The following table shows the por_hnf_class_pocq_arb_weight_ctl higher register bit assignments.

Table 3-1376 por_hnf_por_hnf_class_pocq_arb_weight_ctl (high)

Bits	Field name	Description	Type	Reset
63:45	Reserved	Reserved	RO	-
44:40	hnf_pocq_arb_other_weight	Other req weight for scheduling requests from POCQ	RW	5'b00000
39:37	Reserved	Reserved	RO	-
36:32	hnf_pocq_arb_seq_weight	SEQ weight for scheduling requests from POCQ	RW	5'b00000

The following image shows the lower register bit assignments.

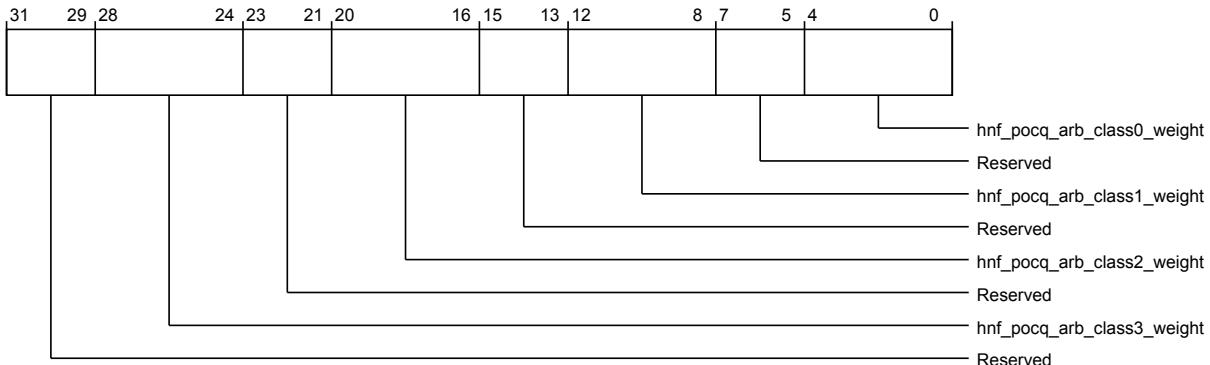


Figure 3-1357 por_hnf_por_hnf_class_pocq_arb_weight_ctl (low)

The following table shows the por_hnf_class_pocq_arb_weight_ctl lower register bit assignments.

Table 3-1377 por_hnf_por_hnf_class_pocq_arb_weight_ctl (low)

Bits	Field name	Description	Type	Reset
31:29	Reserved	Reserved	RO	-
28:24	hnf_pocq_arb_class3_weight	Class3 weight for scheduling requests from POCQ	RW	5'b00000
23:21	Reserved	Reserved	RO	-
20:16	hnf_pocq_arb_class2_weight	Class2 weight for scheduling requests from POCQ	RW	5'b00000
15:13	Reserved	Reserved	RO	-

Table 3-1377 por_hnf_por_hnf_class_pocq_arb_weight_ctl (low) (continued)

Bits	Field name	Description	Type	Reset
12:8	hnf_pocq_arb_class1_weight	Class1 weight for scheduling requests from POCQ	RW	5'b00000
7:5	Reserved	Reserved	RO	-
4:0	hnf_pocq_arb_class0_weight	Class0 weight for scheduling requests from POCQ	RW	5'b00000

por_hnf_class_retry_weight_ctl

Per Class weight controls for Retry Credit grant.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1058

Register reset 64'b0

Usage constraints	Only accessible by secure accesses.
Secure group	por_hnf_secure_register_groups_override.qos

The following section contains the high level architecture of our system.

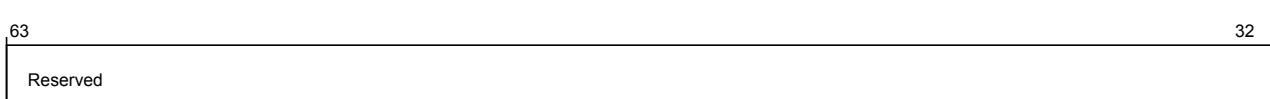


Figure 3-1358 nor hnf nor hnf class retry weight ctrl (high)

The following table shows the nor_hnf, class, retry, weight, ctl higher register bit assignments

Table 3-1378 por hnf por hnf class retry weight ctl (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments

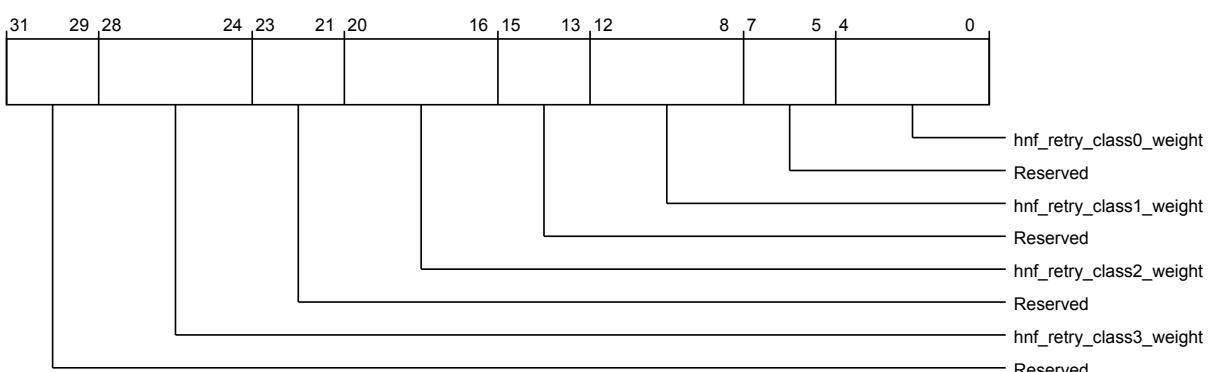


Figure 3-1359 por hnf por hnf class retry weight ctl (low)

The following table shows the por_hnf_class_retry_weight_ctl lower register bit assignments.

Table 3-1379 por_hnf_por_hnf_class_retry_weight_ctl (low)

Bits	Field name	Description	Type	Reset
31:29	Reserved	Reserved	RO	-
28:24	hnf_retry_class3_weight	Overall Class3 weight for credit grant arbitration	RW	5'b00000
23:21	Reserved	Reserved	RO	-
20:16	hnf_retry_class2_weight	Overall Class2 weight for credit grant arbitration	RW	5'b00000
15:13	Reserved	Reserved	RO	-
12:8	hnf_retry_class1_weight	Overall Class1 weight for credit grant arbitration	RW	5'b00000
7:5	Reserved	Reserved	RO	-
4:0	hnf_retry_class0_weight	Overall Class0 weight for credit grant arbitration	RW	5'b00000

por_hnf_rnf_class_weight_0-63

This register repeats 63 times. It parametrized by the index from 0 to 63. RN-F weight registers for static credit grant for all classes

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h1060 + {[0, 1, 2, .., 62, 63]}*8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

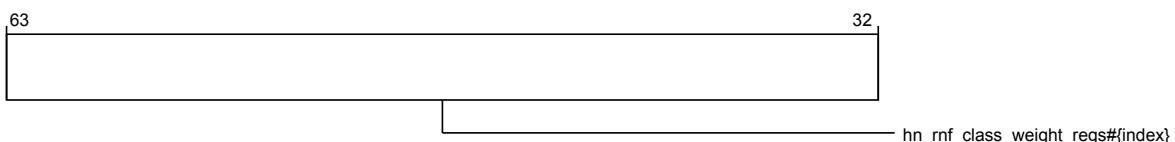


Figure 3-1360 por_hnf_por_hnf_rnf_class_weight_0-63 (high)

The following table shows the por_hnf_rnf_class_weight_0-63 higher register bit assignments.

Table 3-1380 por_hnf_por_hnf_rnf_class_weight_0-63 (high)

Bits	Field name	Description	Type	Reset
63:32	hn_rnf_class_weight_regs#{index}	For each RN-F, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for RNF LDID 0 regs0[12:8]: Weights for RNF LDID 1 regs0[20:16]: Weights for RNF LDID 2 on so forth.	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

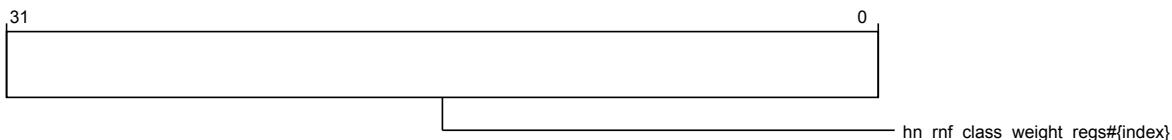


Figure 3-1361 por_hnf_por_hnf_rnf_class_weight_0-63 (low)

The following table shows the por_hnf_rnf_class_weight_0-63 lower register bit assignments.

Table 3-1381 por_hnf_por_hnf_rnf_class_weight_0-63 (low)

Bits	Field name	Description	Type	Reset
31:0	hn_rnf_class_weight_regs#{index}	For each RN-F, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for RNF LDID 0 regs0[12:8]: Weights for RNF LDID 1 regs0[20:16]: Weights for RNF LDID 2 on so forth.	RW	64'h0000000000000000

por_hnf_rni_class_weight_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. RN-I weight registers for static credit grant for all classes

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1260 + {[0, 1, 2, .., 14, 15]}*8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

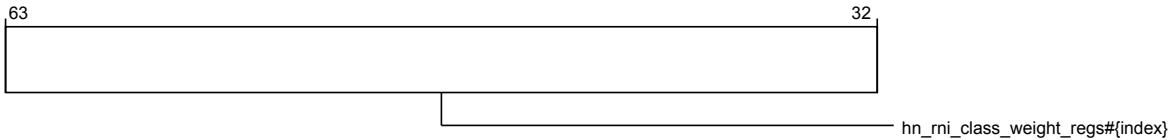


Figure 3-1362 por_hnf_por_hnf_rni_class_weight_0-15 (high)

The following table shows the por_hnf_rni_class_weight_0-15 higher register bit assignments.

Table 3-1382 por_hnf_por_hnf_rni_class_weight_0-15 (high)

Bits	Field name	Description	Type	Reset
63:32	hn_rni_class_weight_regs#{index}	For each RN-I, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for RNI LDID 0 regs0[12:8]: Weights for RNI LDID 1 regs0[20:16]: Weights for RNI LDID 2 on so forth.	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

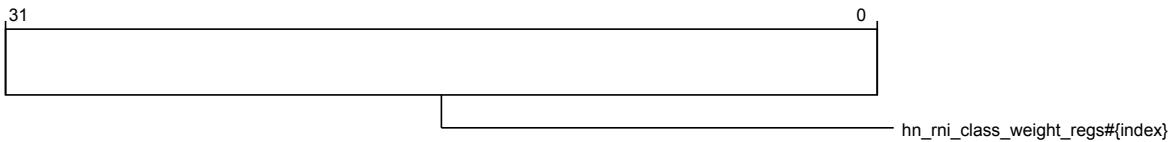


Figure 3-1363 por_hnf_por_hnf_rni_class_weight_0-15 (low)

The following table shows the por_hnf_rni_class_weight_0-15 lower register bit assignments.

Table 3-1383 por_hnf_por_hnf_rni_class_weight_0-15 (low)

Bits	Field name	Description	Type	Reset
31:0	hn_rni_class_weight_regs#{index}	For each RN-I, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for RNI LDID 0 regs0[12:8]: Weights for RNI LDID 1 regs0[20:16]: Weights for RNI LDID 2 on so forth.	RW	64'h0000000000000000

por_hnf_rnd_class_weight_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. RN-D weight registers for static credit grant for all classes

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h12E0 + # {[0, 1, 2, .., 14, 15]} * 8

Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

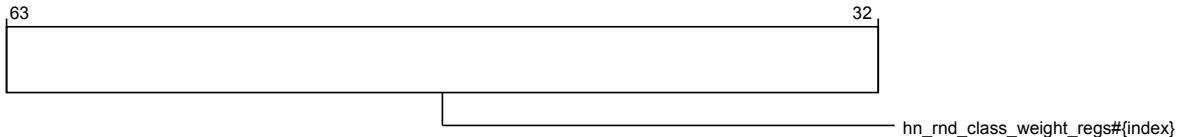


Figure 3-1364 por_hnf_por_hnf_rnd_class_weight_0-15 (high)

The following table shows the por_hnf_rnd_class_weight_0-15 higher register bit assignments.

Table 3-1384 por_hnf_por_hnf_rnd_class_weight_0-15 (high)

Bits	Field name	Description	Type	Reset
63:32	hn_rnd_class_weight_regs#{index}	For each RN-D, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for RND LDID 0 regs0[12:8]: Weights for RND LDID 1 regs0[20:16]: Weights for RND LDID 2 on so forth.	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

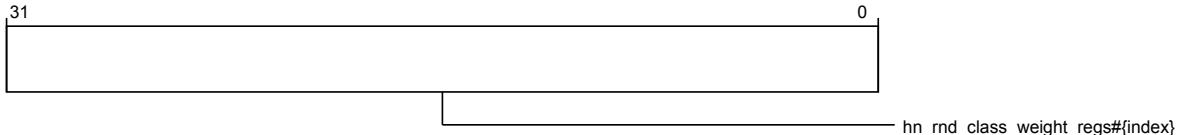


Figure 3-1365 por_hnf_por_hnf_rnd_class_weight_0-15 (low)

The following table shows the por_hnf_rnd_class_weight_0-15 lower register bit assignments.

Table 3-1385 por_hnf_por_hnf_rnd_class_weight_0-15 (low)

Bits	Field name	Description	Type	Reset
31:0	hn_rnd_class_weight_regs#{index}	For each RN-D, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for RND LDID 0 regs0[12:8]: Weights for RND LDID 1 regs0[20:16]: Weights for RND LDID 2 on so forth.	RW	64'h0000000000000000

por_hnf_cxha_class_weight_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. CXHA weight registers for static credit grant for all classes

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h1360 + # {[0, 1, 2, .., 14, 15]} * 8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.qos

The following image shows the higher register bit assignments.

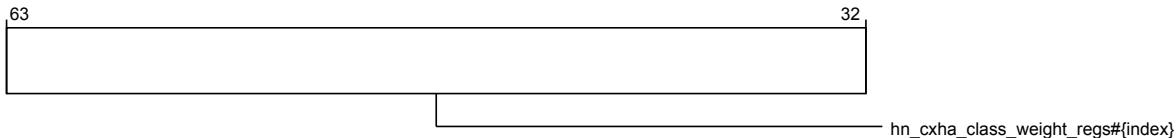


Figure 3-1366 por_hnf_por_hnf_cxha_class_weight_0-15 (high)

The following table shows the por_hnf_cxha_class_weight_0-15 higher register bit assignments.

Table 3-1386 por_hnf_por_hnf_cxha_class_weight_0-15 (high)

Bits	Field name	Description	Type	Reset
63:32	hn(cxha_class_weight_regs#{index})	For each CXHA, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for CXHA LDID 0 regs0[12:8]: Weights for CXHA LDID 1 regs0[20:16]: Weights for CXHA LDID 2 on so forth.	RW	64'h0000000000000000

The following image shows the lower register bit assignments.

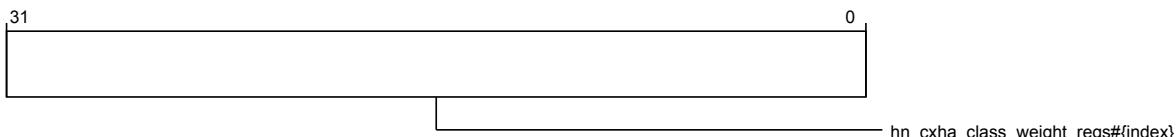


Figure 3-1367 por_hnf_por_hnf_cxha_class_weight_0-15 (low)

The following table shows the por_hnf_cxha_class_weight_0-15 lower register bit assignments.

Table 3-1387 por_hnf_por_hnf_cxha_class_weight_0-15 (low)

Bits	Field name	Description	Type	Reset
31:0	hn(cxha_class_weight_regs#{index})	For each CXHA, 5 bit weight register and 3 bits reserved for future. regs0[4:0]: Weights for CXHA LDID 0 regs0[12:8]: Weights for CXHA LDID 1 regs0[20:16]: Weights for CXHA LDID 2 on so forth.	RW	64'h0000000000000000

por_hnf_partner_scratch_reg0

Partner scratch register 0

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hFE0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.partner_scratch_override

The following image shows the higher register bit assignments.

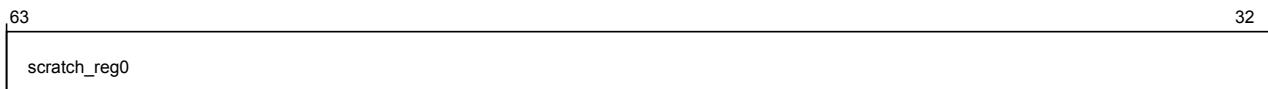


Figure 3-1368 por_hnf_por_hnf_partner_scratch_reg0 (high)

The following table shows the por_hnf_partner_scratch_reg0 higher register bit assignments.

Table 3-1388 por_hnf_por_hnf_partner_scratch_reg0 (high)

Bits	Field name	Description	Type	Reset
63:32	scratch_reg0	64 bit scratch register 0 with read/write access	RW	64'h00000000

The following image shows the lower register bit assignments.

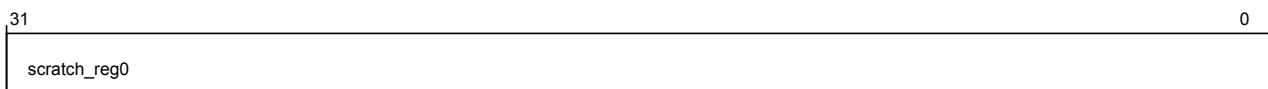


Figure 3-1369 por_hnf_por_hnf_partner_scratch_reg0 (low)

The following table shows the por_hnf_partner_scratch_reg0 lower register bit assignments.

Table 3-1389 por_hnf_por_hnf_partner_scratch_reg0 (low)

Bits	Field name	Description	Type	Reset
31:0	scratch_reg0	64 bit scratch register 0 with read/write access	RW	64'h00000000

por_hnf_partner_scratch_reg1

Partner scratch register 1

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'hFE8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.partner_scratch_override

The following image shows the higher register bit assignments.

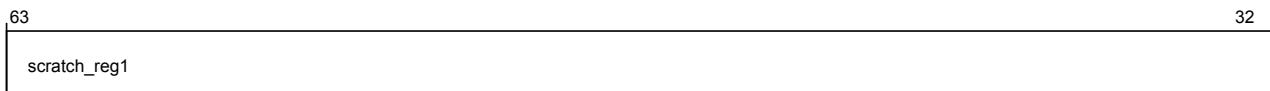


Figure 3-1370 por_hnf_por_hnf_partner_scratch_reg1 (high)

The following table shows the por_hnf_partner_scratch_reg1 higher register bit assignments.

Table 3-1390 por_hnf_por_hnf_partner_scratch_reg1 (high)

Bits	Field name	Description	Type	Reset
63:32	scratch_reg1	64 bit scratch register 1 with read/write access	RW	64'h00000000

The following image shows the lower register bit assignments.



Figure 3-1371 por_hnf_por_hnf_partner_scratch_reg1 (low)

The following table shows the por_hnf_partner_scratch_reg1 lower register bit assignments.

Table 3-1391 por_hnf_por_hnf_partner_scratch_reg1 (low)

Bits	Field name	Description	Type	Reset
31:0	scratch_reg1	64 bit scratch register 1 with read/write access	RW	64'h00000000

por_hnf_cfg_slcsf_dbgrd

Controls access modes for SLC tag, SLC data, and SF tag debug read.

Its characteristics are:

Type	WO
Register width (Bits)	64
Address offset	16'hB80
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.slcsf_dbgrd

The following image shows the higher register bit assignments.

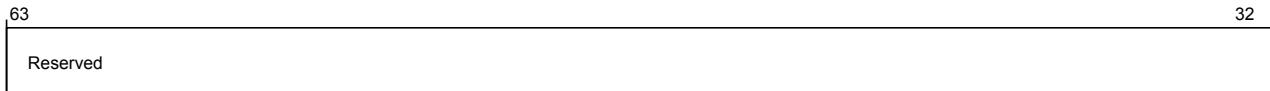


Figure 3-1372 por_hnf_por_hnf_cfg_slcsf_dbgrd (high)

The following table shows the por_hnf_cfg_slcsf_dbgrd higher register bit assignments.

Table 3-1392 por_hnf_por_hnf_cfg_slcsf_dbgrd (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

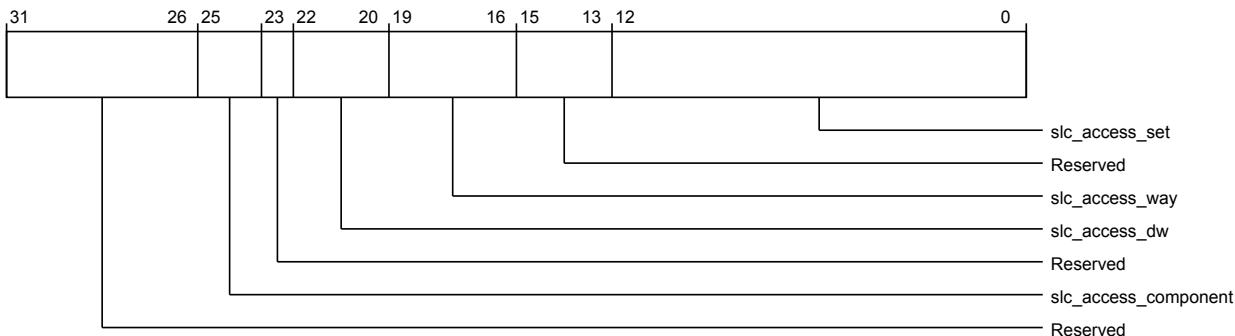


Figure 3-1373 por_hnf_por_hnf_cfg_slcsf_dbgrd (low)

The following table shows the por_hnf_cfg_slcsf_dbgrd lower register bit assignments.

Table 3-1393 por_hnf_por_hnf_cfg_slcsf_dbgrd (low)

Bits	Field name	Description	Type	Reset
31:26	Reserved	Reserved	RO	-
25:24	slc_access_component	Specifies SLC/SF array debug read 2'b01: SLC data read 2'b10: SLC tag read 2'b11: SF tag read	WO	2'b00

Table 3-1393 por_hnf_por_hnf_cfg_slcsf_dbgrd (low) (continued)

Bits	Field name	Description	Type	Reset
23	Reserved	Reserved	RO	-
22:20	slc_access_dw	64-bit chunk address for SLC data debug read access	WO	3'h0
19:16	slc_access_way	Way address for SLC/SF debug read access	WO	4'h0
15:13	Reserved	Reserved	RO	-
12:0	slc_access_set	Set address for SLC/SF debug read access	WO	13'h0

por_hnf_slc_cache_access_slc_tag

Contains SLC tag debug read data bits [63:0]

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hB88

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slcsf_dbgrd

The following image shows the higher register bit assignments.

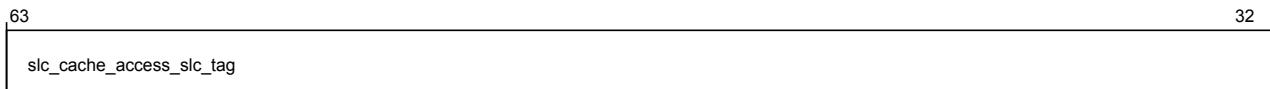


Figure 3-1374 por_hnf_por_hnf_slc_cache_access_slc_tag (high)

The following table shows the por_hnf_slc_cache_access_slc_tag higher register bit assignments.

Table 3-1394 por_hnf_por_hnf_slc_cache_access_slc_tag (high)

Bits	Field name	Description	Type	Reset
63:32	slc_cache_access_slc_tag	SLC tag debug read data	RO	64'h0

The following image shows the lower register bit assignments.

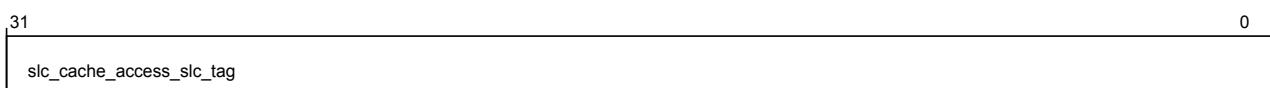


Figure 3-1375 por_hnf_por_hnf_slc_cache_access_slc_tag (low)

The following table shows the por_hnf_slc_cache_access_slc_tag lower register bit assignments.

Table 3-1395 por_hnf_por_hnf_slc_cache_access_slc_tag (low)

Bits	Field name	Description	Type	Reset
31:0	slc_cache_access_slc_tag	SLC tag debug read data	RO	64'h0

por_hnf_slc_cache_access_slc_tag1

Contains SLC tag debug read data bits [127:64] when present

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hB90

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slc_sf_dbgrd

The following image shows the higher register bit assignments.

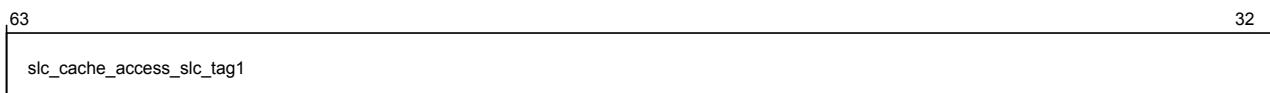


Figure 3-1376 por_hnf_por_hnf_slc_cache_access_slc_tag1 (high)

The following table shows the por_hnf_slc_cache_access_slc_tag1 higher register bit assignments.

Table 3-1396 por_hnf_por_hnf_slc_cache_access_slc_tag1 (high)

Bits	Field name	Description	Type	Reset
63:32	slc_cache_access_slc_tag1	SLC tag debug read data	RO	64'h0

The following image shows the lower register bit assignments.

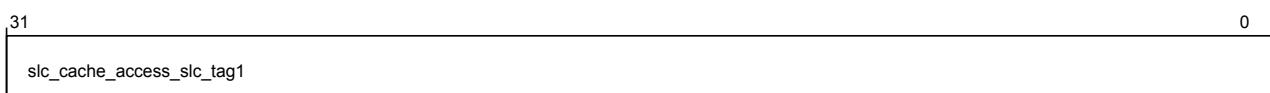


Figure 3-1377 por_hnf_por_hnf_slc_cache_access_slc_tag1 (low)

The following table shows the por_hnf_slc_cache_access_slc_tag1 lower register bit assignments.

Table 3-1397 por_hnf_por_hnf_slc_cache_access_slc_tag1 (low)

Bits	Field name	Description	Type	Reset
31:0	slc_cache_access_slc_tag1	SLC tag debug read data	RO	64'h0

por_hnf_slc_cache_access_slc_data

Contains SLC data RAM debug read data.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'hB98
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.slcsf_dbgrd

The following image shows the higher register bit assignments.

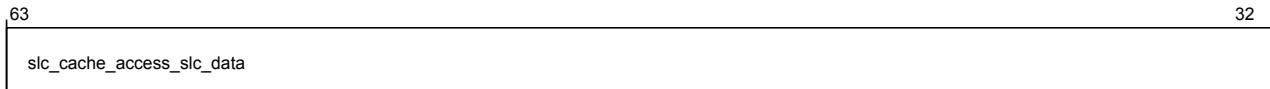


Figure 3-1378 por_hnf_por_hnf_slc_cache_access_slc_data (high)

The following table shows the por_hnf_slc_cache_access_slc_data higher register bit assignments.

Table 3-1398 por_hnf_por_hnf_slc_cache_access_slc_data (high)

Bits	Field name	Description	Type	Reset
63:32	slc_cache_access_slc_data	SLC data RAM debug read data	RO	64'h0

The following image shows the lower register bit assignments.

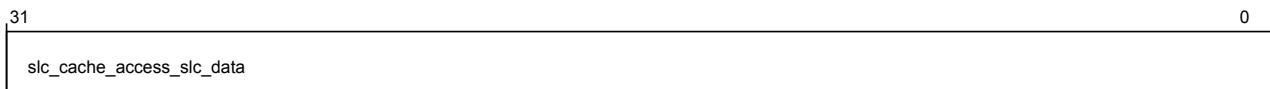


Figure 3-1379 por_hnf_por_hnf_slc_cache_access_slc_data (low)

The following table shows the por_hnf_slc_cache_access_slc_data lower register bit assignments.

Table 3-1399 por_hnf_por_hnf_slc_cache_access_slc_data (low)

Bits	Field name	Description	Type	Reset
31:0	slc_cache_access_slc_data	SLC data RAM debug read data	RO	64'h0

por_hnf_slc_cache_access_slc_mte_tag

Contains MTE Tag data for the corresponding SLC data RAM debug read.

Its characteristics are:

Type	RO
Register width (Bits)	64
Address offset	16'hBC0
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slcsf_dbgrd

The following image shows the higher register bit assignments.

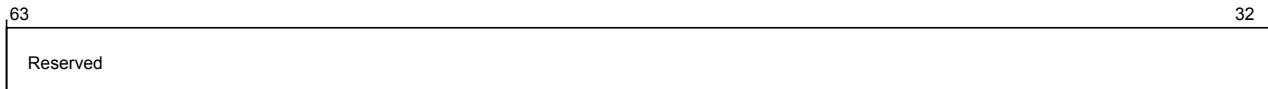


Figure 3-1380 por_hnf_por_hnf_slc_cache_access_slc_mte_tag (high)

The following table shows the por_hnf_slc_cache_access_slc_mte_tag higher register bit assignments.

Table 3-1400 por_hnf_por_hnf_slc_cache_access_slc_mte_tag (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

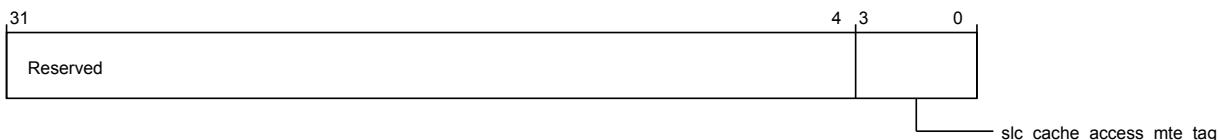


Figure 3-1381 por_hnf_por_hnf_slc_cache_access_slc_mte_tag (low)

The following table shows the por_hnf_slc_cache_access_slc_mte_tag lower register bit assignments.

Table 3-1401 por_hnf_por_hnf_slc_cache_access_slc_mte_tag (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	slc_cache_access_mte_tag	SLC MTE TAG corresponding to data RAM debug read data (128bit chunk of data)	RO	4'h0

por_hnf_slc_cache_access_sf_tag

Contains SF tag debug read data. Bits[63:0]

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hBA0

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slcsf_dbgrd

The following image shows the higher register bit assignments.

63

32

slc_cache_access_sf_tag

Figure 3-1382 por_hnf_por_hnf_slc_cache_access_sf_tag (high)

The following table shows the por_hnf_slc_cache_access_sf_tag higher register bit assignments.

Table 3-1402 por_hnf_por_hnf_slc_cache_access_sf_tag (high)

Bits	Field name	Description	Type	Reset
63:32	slc_cache_access_sf_tag	SF tag debug read data	RO	64'h0

The following image shows the lower register bit assignments.

31

0

slc_cache_access_sf_tag

Figure 3-1383 por_hnf_por_hnf_slc_cache_access_sf_tag (low)

The following table shows the por_hnf_slc_cache_access_sf_tag lower register bit assignments.

Table 3-1403 por_hnf_por_hnf_slc_cache_access_sf_tag (low)

Bits	Field name	Description	Type	Reset
31:0	slc_cache_access_sf_tag	SF tag debug read data	RO	64'h0

por_hnf_slc_cache_access_sf_tag1

Contains SF tag debug read data bits [127:64], when present in SF Tag

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hBA8

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slc_sf_dbgrd

The following image shows the higher register bit assignments.

63

32

slc_cache_access_sf_tag1

Figure 3-1384 por_hnf_por_hnf_slc_cache_access_sf_tag1 (high)

The following table shows the por_hnf_slc_cache_access_sf_tag1 higher register bit assignments.

Table 3-1404 por_hnf_por_hnf_slc_cache_access_sf_tag1 (high)

Bits	Field name	Description	Type	Reset
63:32	slc_cache_access_sf_tag1	SF tag debug read data	RO	64'h0

The following image shows the lower register bit assignments.

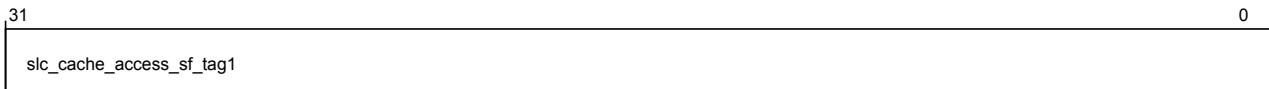


Figure 3-1385 por_hnf_por_hnf_slc_cache_access_sf_tag1 (low)

The following table shows the por_hnf_slc_cache_access_sf_tag1 lower register bit assignments.

Table 3-1405 por_hnf_por_hnf_slc_cache_access_sf_tag1 (low)

Bits	Field name	Description	Type	Reset
31:0	slc_cache_access_sf_tag1	SF tag debug read data	RO	64'h0

por_hnf_slc_cache_access_sf_tag2

Contains SF tag debug read data bits [128:191], when present in SF Tag

Its characteristics are:

Type RO

Register width (Bits) 64

Address offset 16'hBB0

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.slcsf_dbgrd

The following image shows the higher register bit assignments.



Figure 3-1386 por_hnf_por_hnf_slc_cache_access_sf_tag2 (high)

The following table shows the por_hnf_slc_cache_access_sf_tag2 higher register bit assignments.

Table 3-1406 por_hnf_por_hnf_slc_cache_access_sf_tag2 (high)

Bits	Field name	Description	Type	Reset
63:32	slc_cache_access_sf_tag2	SF tag debug read data	RO	64'h0

The following image shows the lower register bit assignments.

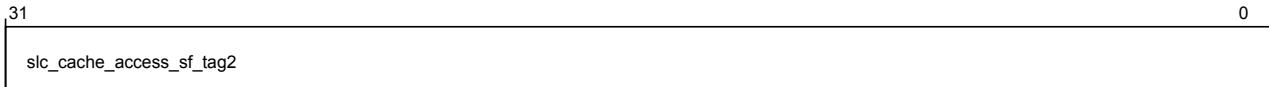


Figure 3-1387 por_hnf_por_hnf_slc_cache_access_sf_tag2 (low)

The following table shows the por_hnf_slc_cache_access_sf_tag2 lower register bit assignments.

Table 3-1407 por_hnf_por_hnf_slc_cache_access_sf_tag2 (low)

Bits	Field name	Description	Type	Reset
31:0	slc_cache_access_sf_tag2	SF tag debug read data	RO	64'h0

por_hnf_pmu_event_sel

Specifies the PMU event to be counted.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2000

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.

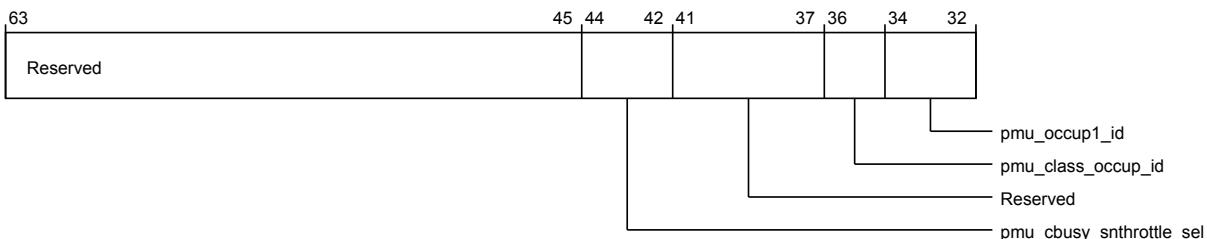


Figure 3-1388 por_hnf_por_hnf_pmu_event_sel (high)

The following table shows the por_hnf_pmu_event_sel higher register bit assignments.

Table 3-1408 por_hnf_por_hnf_pmu_event_sel (high)

Bits	Field name	Description	Type	Reset
63:45	Reserved	Reserved	RO	-
44:42	pmu_cbusy_snthrottle_sel	Filter for selecting specific SN throttle type 3'b000: All SN types throttled 3'b001: SN Group 0 Reads 3'b010: SN Group 0 Non-Reads 3'b011: SN Group 1 Reads 3'b100: SN Group 1 Non-Reads 3'b101: All SN Reads 3'b110: All SN Non-Reads	RW	3'h0
41:37	Reserved	Reserved	RO	-
36:35	pmu_class_occup_id	HN-F PMU Class select 2'b00: Class 0 selected 2'b01: Class 1 selected 2'b10: Class 2 selected 2'b11: Class 3 selected	RW	2'h0
34:32	pmu_occup1_id	HN-F PMU occupancy 1 select 3'b000: All occupancy selected 3'b001: Read requests 3'b010: Write requests 3'b011: Atomic operation requests 3'b100: Stash requests	RW	3'h0

The following image shows the lower register bit assignments.

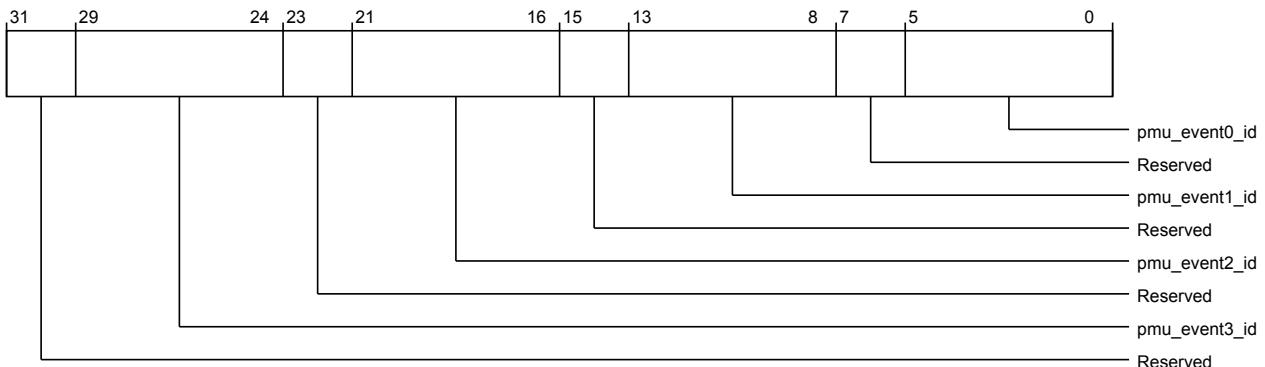


Figure 3-1389 por_hnf_por_hnf_pmu_event_sel (low)

The following table shows the por_hnf_pmu_event_sel lower register bit assignments.

Table 3-1409 por_hnf_por_hnf_pmu_event_sel (low)

Bits	Field name	Description	Type	Reset
31:30	Reserved	Reserved	RO	-
29:24	pmu_event3_id	HN-F PMU Event 3 select; see pmu_event0_id for encodings	RW	6'h00
23:22	Reserved	Reserved	RO	-
21:16	pmu_event2_id	HN-F PMU Event 2 select; see pmu_event0_id for encodings	RW	6'h00
15:14	Reserved	Reserved	RO	-
13:8	pmu_event1_id	HN-F PMU Event 1 select; see pmu_event0_id for encodings	RW	6'h00

Table 3-1409 por_hnf_por_hnf_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
7:6	Reserved	Reserved	RO	-

Table 3-1409 por_hnf_por_hnf_pmu_event_sel (low) (continued)

Bits	Field name	Description	Type	Reset
5:0	pmu_event0_id	<p>HN-F PMU Event 0 select</p> <p>6'h00: No event</p> <p>6'h01: PMU_HN_CACHE_MISS_EVENT; counts total cache misses in first lookup result (high priority)</p> <p>6'h02: PMU_HN_SLCSF_CACHE_ACCESS_EVENT; counts number of cache accesses in first access (high priority)</p> <p>6'h03: PMU_HN_CACHE_FILL_EVENT; counts total allocations in HN SLC (all cache line allocations to SLC)</p> <p>6'h04: PMU_HN_POCQ_RETRY_EVENT; counts number of retried requests</p> <p>6'h05: PMU_HN_POCQ_REQS_RECV_EVENT; counts number of requests received by HN</p> <p>6'h06: PMU_HN_SF_HIT_EVENT; counts number of SF hits</p> <p>6'h07: PMU_HN_SF_EVICTIONS_EVENT; counts number of SF eviction cache invalidations initiated</p> <p>6'h08: PMU_HN_DIR_SNOOPS_SENT_EVENT; counts number of directed snoops sent (not including SF back invalidation)</p> <p>6'h09: PMU_HN_BRD_SNOOPS_SENTEVENT; counts number of multicast snoops send (not including SF back invalidation)</p> <p>6'h0A: PMU_HN_SLC_EVICTION_EVENT; counts number of SLC evictions (dirty only)</p> <p>6'h0B: PMU_HN_SLC_FILL_INVALID_WAY_EVENT; counts number of SLC fills to an invalid way</p> <p>6'h0C: PMU_HN_MC_RETRIES_EVENT; counts number of retried transactions by the MC</p> <p>6'h0D: PMU_HN_MC_REQS_EVENT; counts number of requests sent to MC</p> <p>6'h0E: PMU_HN_QOS_HH_RETRY_EVENT; counts number of times a HighHigh priority request is protocol retried at the HN-F</p> <p>6'h0F: PMU_HN_POCQ_OCCUPANCY_EVENT; counts the POCQ occupancy in HN-F; occupancy filtering is programmed in pmu_occup1_id</p> <p>6'h10: PMU_HN_POCQ_ADDRHAZ_EVENT; counts number of POCQ address hazards upon allocation</p> <p>6'h11: PMU_HN_POCQ_ATOMICS_ADDRHAZ_EVENT; counts number of POCQ address hazards upon allocation for atomic operations</p> <p>6'h12: PMU_HN_LD_ST_SWP_ADQ_FULL_EVENT; counts number of times ADQ is full for Ld/St/SWP type atomic operations while POCQ has pending operations</p> <p>6'h13: PMU_HN_CMP_ADQ_FULL_EVENT; counts number of times ADQ is full for CMP type atomic operations while POCQ has pending operations</p> <p>6'h14: PMU_HN_TXDAT_STALL_EVENT; counts number of times HN-F has a pending TXDAT flit but no credits to upload</p> <p>6'h15: PMU_HN_TXRSP_STALL_EVENT; counts number of times HN-F has a pending TXRSP flit but no credits to upload</p> <p>6'h16: PMU_HN_SEQ_FULL_EVENT; counts number of times requests are replayed in SLC pipe due to SEQ being full</p> <p>6'h17: PMU_HN_SEQ_HIT_EVENT; counts number of times a request in SLC hit a pending SF eviction in SEQ</p>	RW	6'h00
102308_0000_01_en		<p>Copyright © 2020 Arm Limited or its affiliates. All rights reserved.</p> <p>Confidential</p> <p>multicast/SF back invalidation</p> <p>6'h19: PMU_HN_SEBL_DIR_SNP_SFNT_EVENT; counts number of times directed snoops</p>		3-1199

por_hnf_pmu_mpam_sel

Specifies details of MPAM event to be counted

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2008

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1390 por_hnf_por_hnf_pmu_mpam_sel (high)

The following table shows the por_hnf_pmu_mpam_sel higher register bit assignments.

Table 3-1410 por_hnf_por_hnf_pmu_mpam_sel (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

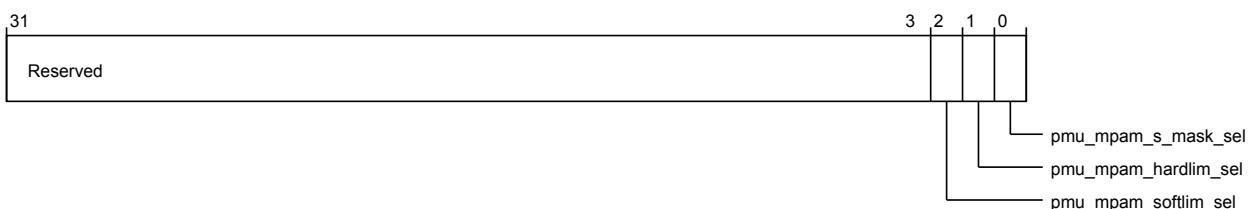


Figure 3-1391 por_hnf_por_hnf_pmu_mpam_sel (low)

The following table shows the por_hnf_pmu_mpam_sel lower register bit assignments.

Table 3-1411 por_hnf_por_hnf_pmu_mpam_sel (low)

Bits	Field name	Description	Type	Reset
31:3	Reserved	Reserved	RO	-
2	pmu_mpam_softlim_sel	When set, HN-F PMU MPAM Softlimit count is filtered for specific PARTIDs 1'b0: PMU Softlimit count is total for all PARDIDs. 1'b1: PMU Softlimit count is only for PARDIDs indicated in filter register	RW	1'b0

Table 3-1411 por_hnf_por_hnf_pmu_mpam_sel (low) (continued)

Bits	Field name	Description	Type	Reset
1	pmu_mpam_hardlim_sel	When set, HN-F PMU MPAM Hardlimit count is filtered for specific PARTIDs 1'b0: PMU Hardlimit count is total for all PARDIDs. 1'b1: PMU Hardlimit count is only for PARDIDs indicated in fliter register	RW	1'b0
0	pmu_mpam_s_mask_sel	When set, PARTID Mask is used for Secure MPAM PARTID 1'b0: PMU MPAM mask is for Non-Secure MPAMID. 1'b1: PMU MPAM mask is for Secure MPAMID.	RW	1'b0

por_hnf_pmu_mpam_pardid_mask_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Functions as mask for PARTID[#{64*(index+1)-1}:#{64*index}] filter for MPM PMU events

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h2010 + #{8*[0, 1, 2, 3, 4, 5, 6, 7]}

Register reset 64'b0

Usage constraints There are no usage constraints.

The following image shows the higher register bit assignments.



Figure 3-1392 por_hnf_por_hnf_pmu_mpam_pardid_mask_0-7 (high)

The following table shows the por_hnf_pmu_mpam_pardid_mask_0-7 higher register bit assignments.

Table 3-1412 por_hnf_por_hnf_pmu_mpam_pardid_mask_0-7 (high)

Bits	Field name	Description	Type	Reset
63:32	pmu_mpam_partid_mask#{index}	MPAM PMU hardlimit and softlimit mask for PARTID [#{64*(index+1)-1}:#{64*index}] 1'b0: PARTID specified is not counted in PMU count. 1'b1: PARTID specified is counted in PMU count. Note: This mask is used only when por_hnf_pmu_mpam_sel is set for PARTID based counting.	RW	64'b0

The following image shows the lower register bit assignments.

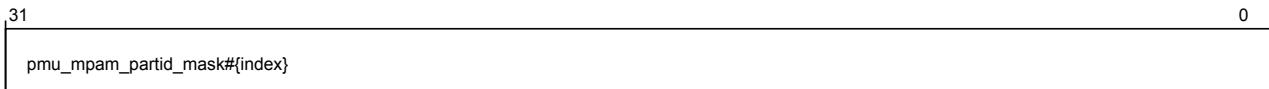


Figure 3-1393 por_hnf_por_hnf_pmu_mpam_pardid_mask_0-7 (low)

The following table shows the por_hnf_pmu_mpam_pardid_mask_0-7 lower register bit assignments.

Table 3-1413 por_hnf_por_hnf_pmu_mpam_pardid_mask_0-7 (low)

Bits	Field name	Description	Type	Reset
31:0	pmu_mpam_partid_mask#{index}	MPAM PMU hardlimit and softlimit mask for PARTID [$\#(64*(index+1)-1):\#(64*index)$] 1'b0: PARTID specified is not counted in PMU count. 1'b1: PARTID specified is counted in PMU count. Note: This mask is used only when por_hnf_pmu_mpam_sel is set for PARTID based counting.	RW	64'b0

por_hnf_rn_cluster_0-63_physid_reg0

This register repeats 63 times. It parametrized by the index from 0 to 63. Configures node IDs for RNs in the system corresponding to each RN ID.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3C00 + {[0, 1, 2, .., 62, 63]}*32

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

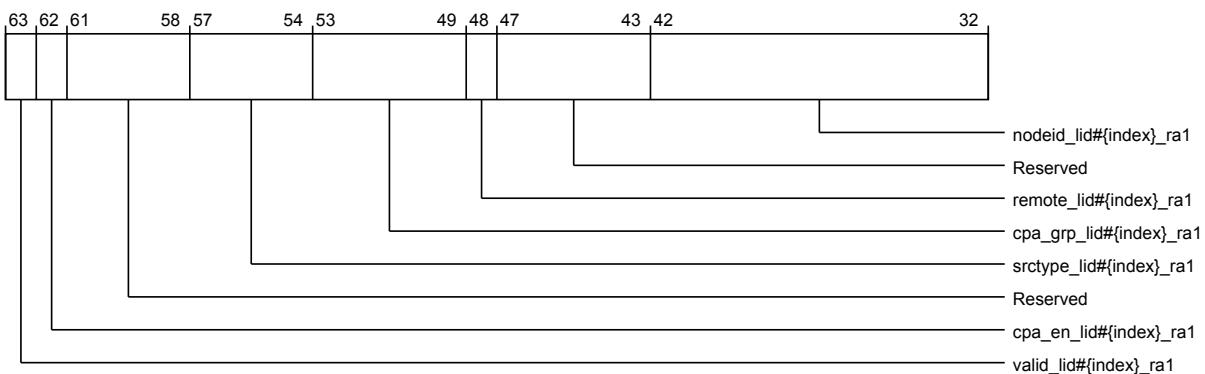


Figure 3-1394 por_hnf_por_hnf_rn_cluster_0-63_physid_reg0 (high)

The following table shows the por_hnf_rn_cluster_0-63_physid_reg0 higher register bit assignments.

Table 3-1414 por_hnf_por_hnf_rn_cluster_0-63_physid_reg0 (high)

Bits	Field name	Description	Type	Reset
63	valid_lid#{index}_ra1	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
62	cpa_en_lid#{index}_ra1	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
61:58	Reserved	Reserved	RO	-
57:54	srtctype_lid#{index}_ra1	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
53:49	cpa_grp_lid#{index}_ra1	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
48	remote_lid#{index}_ra1	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
47:43	Reserved	Reserved	RO	-
42:32	nodeid_lid#{index}_ra1	Specifies the node ID	RW	11'h0

The following image shows the lower register bit assignments.

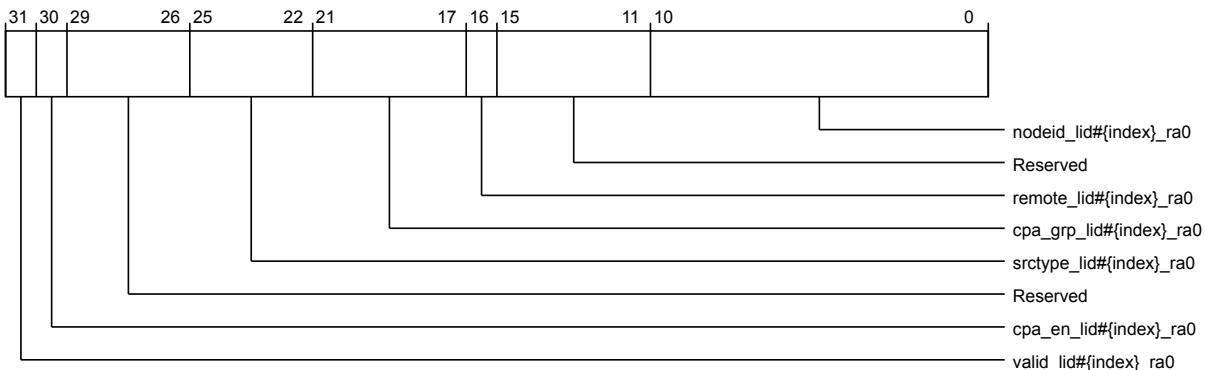


Figure 3-1395 por_hnf_por_hnf_rn_cluster_0-63_physid_reg0 (low)

The following table shows the por hnf rn cluster 0-63 physid reg0 lower register bit assignments.

Table 3-1415 por_hnf_por_hnf_rn_cluster_0-63_physid_reg0 (low)

Bits	Field name	Description	Type	Reset
31	valid_lid#{index}_ra0	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
30	cpa_en_lid#{index}_ra0	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
29:26	Reserved	Reserved	RO	-
25:22	srctype_lid#{index}_ra0	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
21:17	cpa_grp_lid#{index}_ra0	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
16	remote_lid#{index}_ra0	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
15:11	Reserved	Reserved	RO	-
10:0	nodeid_lid#{index}_ra0	Specifies the node ID	RW	11'h0

por_hnf_rn_cluster_64-127_physid_reg0

This register repeats 63 times. It parametrized by the index from 64 to 127. Configures node IDs for RNs in the system corresponding to each RN ID.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3C00 + #{{0, 1, 2, .., 126, 127}}*32

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

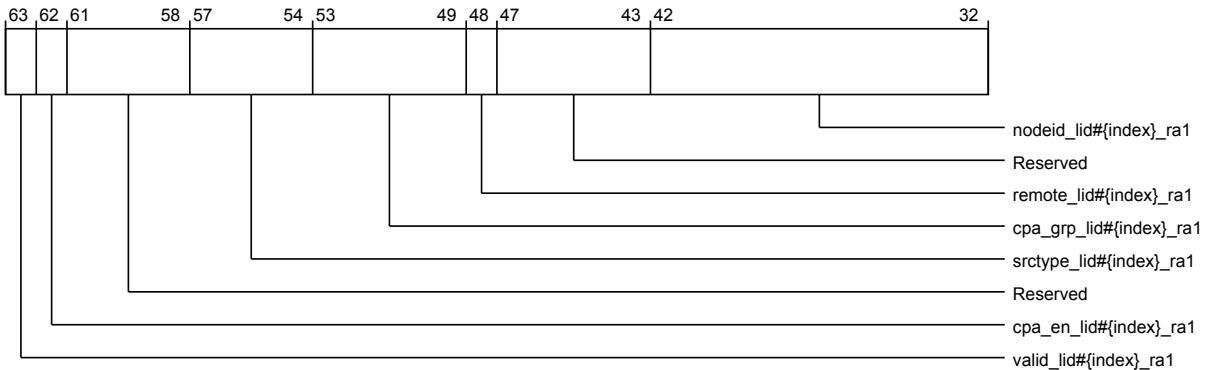


Figure 3-1396 por_hnf_por_hnf_rn_cluster_64-127_physid_reg0 (high)

The following table shows the por hnf rn cluster 64-127 phsids reg0 higher register bit assignments.

Table 3-1416 por_hnf_por_hnf_rn_cluster_64-127_physid_reg0 (high)

Bits	Field name	Description	Type	Reset
63	valid_lid#{index}_ra1	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
62	cpa_en_lid#{index}_ra1	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
61:58	Reserved	Reserved	RO	-
57:54	srctype_lid#{index}_ra1	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
53:49	cpa_grp_lid#{index}_ra1	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
48	remote_lid#{index}_ra1	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
47:43	Reserved	Reserved	RO	-
42:32	nodeid_lid#{index}_ra1	Specifies the node ID	RW	11'h0

The following image shows the lower register bit assignments.

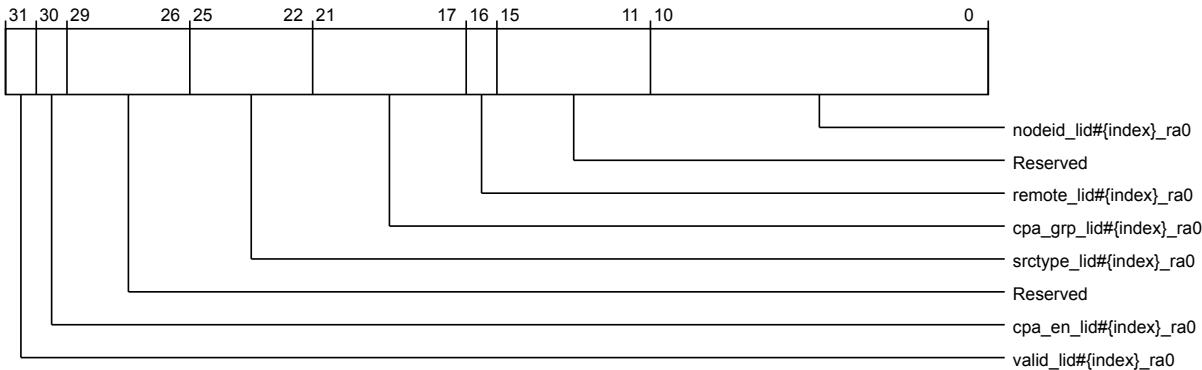


Figure 3-1397 por_hnf_por_hnf_rn_cluster_64-127_physid_reg0 (low)

The following table shows the por_hnf_rn_cluster_64-127_physid_reg0 lower register bit assignments.

Table 3-1417 por_hnf_por_hnf_rn_cluster_64-127_physid_reg0 (low)

Bits	Field name	Description	Type	Reset
31	valid_id#{index}_ra0	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
30	cpa_en_id#{index}_ra0	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
29:26	Reserved	Reserved	RO	-
25:22	srctype_id#{index}_ra0	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
21:17	cpa_grp_id#{index}_ra0	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
16	remote_id#{index}_ra0	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
15:11	Reserved	Reserved	RO	-
10:0	nodeid_id#{index}_ra0	Specifies the node ID	RW	11'h0

por_hnf_rn_cluster_0-127_physid_reg1

This register repeats 127 times. It parametrized by the index from 0 to 127. Configures node IDs for RNs in the system corresponding to each RN ID.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h3C08 + #{{0, 1, 2, .., 126, 127}}*32
Register reset	64'b0
Usage constraints	Only accessible by secure accesses.
Secure group override	por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

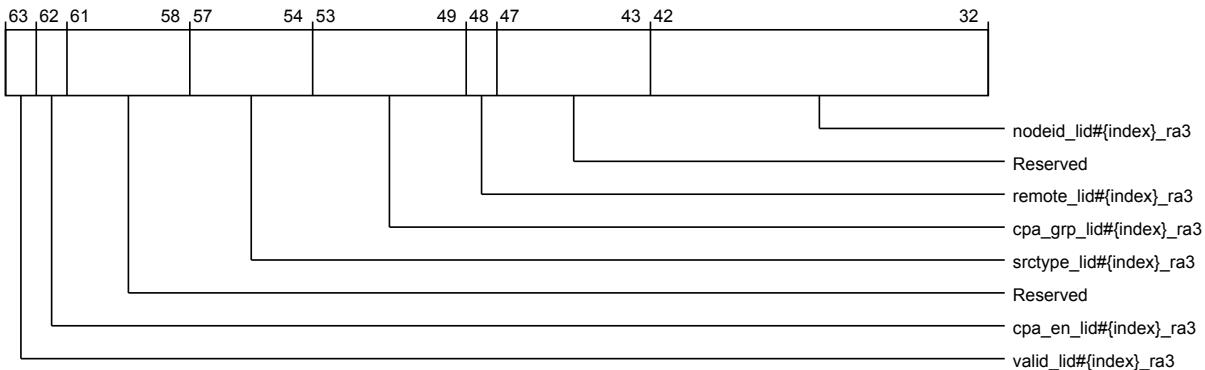


Figure 3-1398 por_hnf_por_hnf_rn_cluster_0-127_physid_reg1 (high)

The following table shows the por_hnf_rn_cluster_0-127_physid_reg1 higher register bit assignments.

Table 3-1418 por_hnf_por_hnf_rn_cluster_0-127_physid_reg1 (high)

Bits	Field name	Description	Type	Reset
63	valid_id#{index}_ra3	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
62	cpa_en_id#{index}_ra3	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
61:58	Reserved	Reserved	RO	-
57:54	srctype_id#{index}_ra3	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
53:49	cpa_grp_id#{index}_ra3	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0

Table 3-1418 por_hnf_por_hnf_rn_cluster_0-127_physid_reg1 (high) (continued)

Bits	Field name	Description	Type	Reset
48	remote_lid#{index}_ra3	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
47:43	Reserved	Reserved	RO	-
42:32	nodeid_lid#{index}_ra3	Specifies the node ID	RW	11'h0

The following image shows the lower register bit assignments.

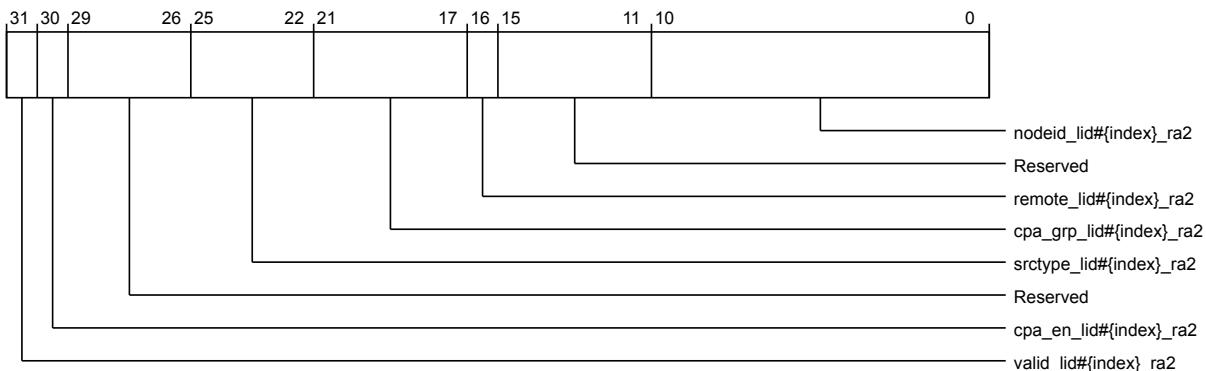


Figure 3-1399 por_hnf_por_hnf_rn_cluster_0-127_physid_reg1 (low)

The following table shows the por_hnf_rn_cluster_0-127_physid_reg1 lower register bit assignments.

Table 3-1419 por_hnf_por_hnf_rn_cluster_0-127_physid_reg1 (low)

Bits	Field name	Description	Type	Reset
31	valid_id#{index}_ra2	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
30	cpa_en_id#{index}_ra2	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
29:26	Reserved	Reserved	RO	-
25:22	srctype_id#{index}_ra2	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000

Table 3-1419 por_hnf_por_hnf_rn_cluster_0-127_physid_reg1 (low) (continued)

Bits	Field name	Description	Type	Reset
21:17	cpa_grp_lid#{index}_ra2	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
16	remote_lid#{index}_ra2	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
15:11	Reserved	Reserved	RO	-
10:0	nodeid_lid#{index}_ra2	Specifies the node ID	RW	11'h0

por_hnf_rn_cluster_0-127_physid_reg2

This register repeats 127 times. It parametrized by the index from 0 to 127. Configures node IDs for RNs in the system corresponding to each RN ID.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3C10 + #[{0, 1, 2, .., 126, 127}]*32

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

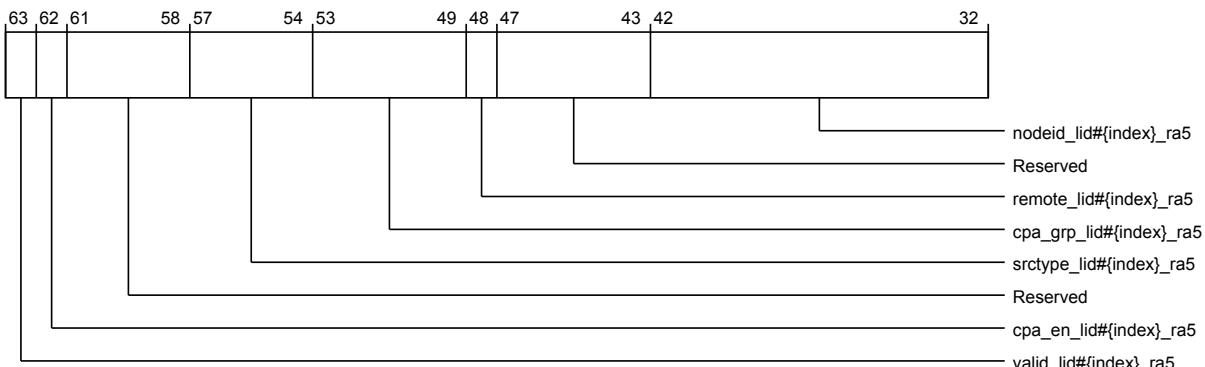


Figure 3-1400 por_hnf_por_hnf_rn_cluster_0-127_physid_reg2 (high)

The following table shows the por_hnf_rn_cluster_0-127_physid_reg2 higher register bit assignments.

Table 3-1420 por_hnf_por_hnf_rn_cluster_0-127_physid_reg2 (high)

Bits	Field name	Description	Type	Reset
63	valid_lid#{index}_ra5	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
62	cpa_en_lid#{index}_ra5	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
61:58	Reserved	Reserved	RO	-
57:54	srtctype_lid#{index}_ra5	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
53:49	cpa_grp_lid#{index}_ra5	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
48	remote_lid#{index}_ra5	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
47:43	Reserved	Reserved	RO	-
42:32	nodeid_lid#{index}_ra5	Specifies the node ID	RW	11'h0

The following image shows the lower register bit assignments.

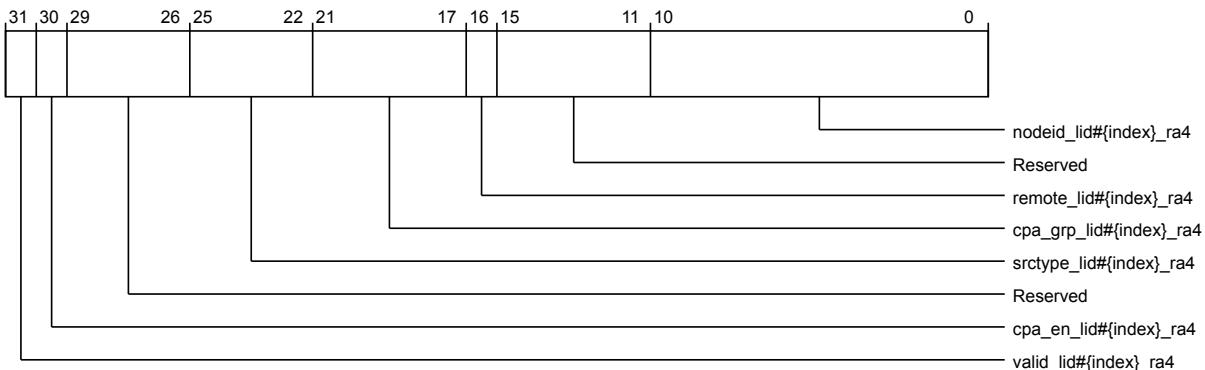


Figure 3-1401 por_hnf_por_hnf_rn_cluster_0-127_physid_reg2 (low)

The following table shows the port assignments for the `hf`, `rm`, `cluster`, `0-127`, `physid`, `reg2`, and `lower register bit` assignments.

Table 3-1421 por_hnf_por_hnf_rn_cluster_0-127_physid_reg2 (low)

Bits	Field name	Description	Type	Reset
31	valid_lid#{index}_ra4	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
30	cpa_en_lid#{index}_ra4	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
29:26	Reserved	Reserved	RO	-
25:22	srctype_lid#{index}_ra4	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
21:17	cpa_grp_lid#{index}_ra4	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
16	remote_lid#{index}_ra4	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
15:11	Reserved	Reserved	RO	-
10:0	nodeid_lid#{index}_ra4	Specifies the node ID	RW	11'h0

por_hnf_rn_cluster_0-127_physid_reg3

This register repeats 127 times. It parametrized by the index from 0 to 127. Configures node IDs for RNs in the system corresponding to each RN ID.

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h3C18 + #{{[0, 1, 2, .., 126, 127]}*32}

Register reset 64'b0

Usage constraints Only accessible by secure accesses.

Secure group override por_hnf_secure_register_groups_override.sam_control

The following image shows the higher register bit assignments.

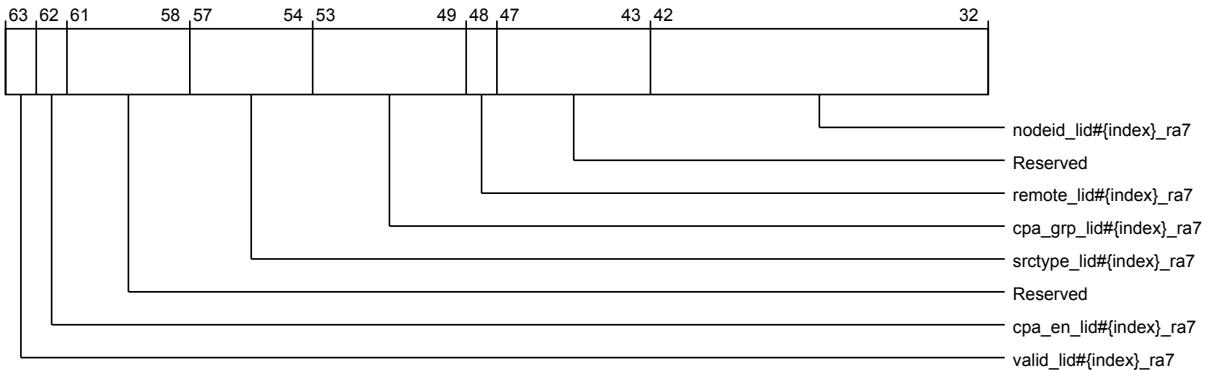


Figure 3-1402 por_hnf_por_hnf_rn_cluster_0-127_physid_reg3 (high)

The following table shows the por_hnf_rn_cluster_0-127_physid_reg3 higher register bit assignments.

Table 3-1422 por_hnf_por_hnf_rn_cluster_0-127_physid_reg3 (high)

Bits	Field name	Description	Type	Reset
63	valid_lid#{index}_ra7	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
62	cpa_en_lid#{index}_ra7	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
61:58	Reserved	Reserved	RO	-
57:54	srctype_lid#{index}_ra7	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
53:49	cpa_grp_lid#{index}_ra7	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
48	remote_lid#{index}_ra7	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
47:43	Reserved	Reserved	RO	-
42:32	nodeid_lid#{index}_ra7	Specifies the node ID	RW	11'h0

The following image shows the lower register bit assignments.

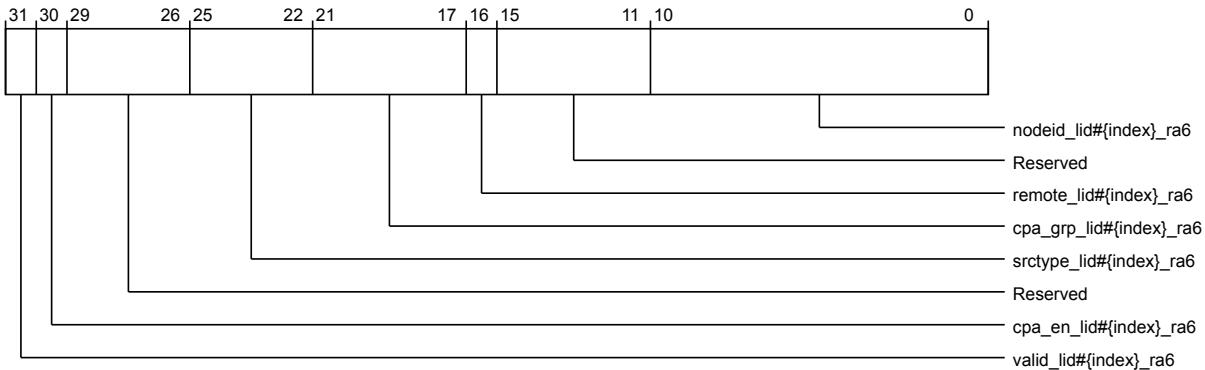


Figure 3-1403 por_hnf_por_hnf_rn_cluster_0-127_physid_reg3 (low)

The following table shows the por_hnf_rn_cluster_0-127_physid_reg3 lower register bit assignments.

Table 3-1423 por_hnf_por_hnf_rn_cluster_0-127_physid_reg3 (low)

Bits	Field name	Description	Type	Reset
31	valid_id#{index}_ra6	Specifies whether the RN is valid 1'b0: RN ID is not valid 1'b1: RN ID is pointing to a valid CHI device	RW	1'h0
30	cpa_en_id#{index}_ra6	Specifies whether the CCIX port aggregation is enabled 1'b0: CPA not enabled 1'b1: CPA enabled	RW	1'h0
29:26	Reserved	Reserved	RO	-
25:22	srctype_id#{index}_ra6	Specifies the CHI source type of the RN 4'b1010: 256 bit CHI-B RN-F 4'b1011: 256 bit CHI-C RN-F 4'b1100: 256 bit CHI-D RN-F 4'b1101: 256 bit CHI-E RN-F Others : Reserved	RW	4'h0000
21:17	cpa_grp_id#{index}_ra6	Specifies CCIX port aggregation group ID(0-31)	RW	5'h0
16	remote_id#{index}_ra6	Specifies whether the RN is remote or local 1'b0: Local RN 1'b1: Remote RN	RW	1'h0
15:11	Reserved	Reserved	RO	-
10:0	nodeid_id#{index}_ra6	Specifies the node ID	RW	11'h0

por_hnf_sam_nonhash_cfg1_memregion_2-63

This register repeats 61 times. It parametrized by the index from 2 to 63. Configures non-hashed memory region #{index} in HN-F SAM.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5000 + {[0, 1, 2, .., 62, 63]}*8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

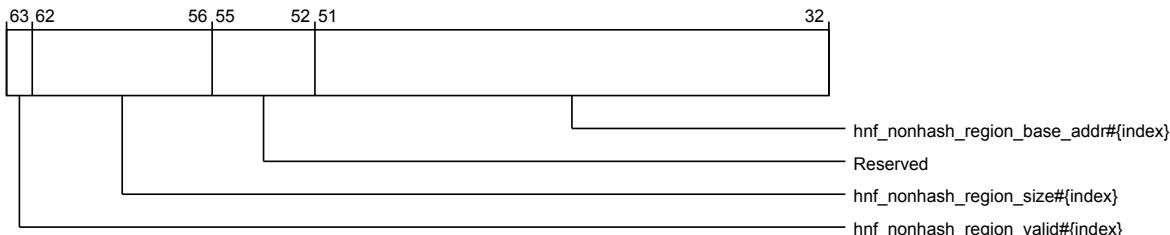


Figure 3-1404 por_hnf_por_hnf_sam_nonhash_cfg1_memregion_2-63 (high)

The following table shows the por_hnf_sam_nonhash_cfg1_memregion_2-63 higher register bit assignments.

Table 3-1424 por_hnf_por_hnf_sam_nonhash_cfg1_memregion_2-63 (high)

Bits	Field name	Description	Type	Reset
63	hnf_nonhash_region_valid#{index}	valid 1'b0: Not valid 1'b1: Valid for memory region comparison	RW	1'b0
62:56	hnf_nonhash_region_size#{index}	Memory region size CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).	RW	7'b0
55:52	Reserved	Reserved	RO	-
51:32	hnf_nonhash_region_base_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0

The following image shows the lower register bit assignments.

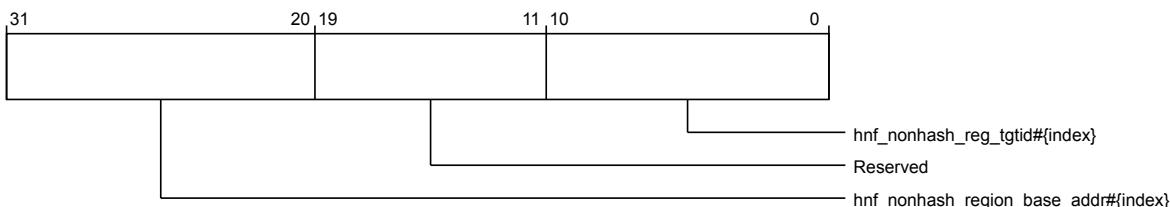


Figure 3-1405 por_hnf_por_hnf_sam_nonhash_cfg1_memregion_2-63 (low)

The following table shows the por_hnf_sam_nonhash_cfg1_memregion_2-63 lower register bit assignments.

Table 3-1425 por_hnf_por_hnf_sam_nonhash_cfg1_memregion_2-63 (low)

Bits	Field name	Description	Type	Reset
31:20	hnf_nonhash_region_base_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0
19:11	Reserved	Reserved	RO	-
10:0	hnf_nonhash_reg_tgtid#{index}	SN TgtID for the non-hashed region	RW	11'h0

por_hnf_sam_nonhash_cfg2_memregion_2-63

This register repeats 61 times. It parametrized by the index from 2 to 63. Configures non-hashed memory region #{index} in HN-F SAM.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5200 + # {[0, 1, 2, .., 62, 63]} * 8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

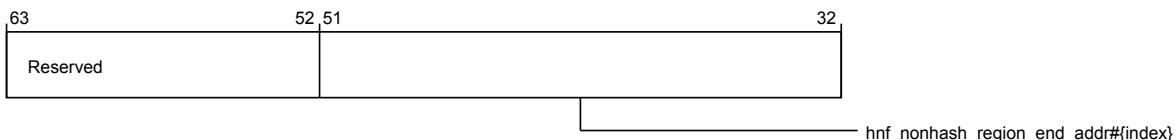


Figure 3-1406 por_hnf_por_hnf_sam_nonhash_cfg2_memregion_2-63 (high)

The following table shows the por_hnf_sam_nonhash_cfg2_memregion_2-63 higher register bit assignments.

Table 3-1426 por_hnf_por_hnf_sam_nonhash_cfg2_memregion_2-63 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	hnf_nonhash_region_end_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0

The following image shows the lower register bit assignments.

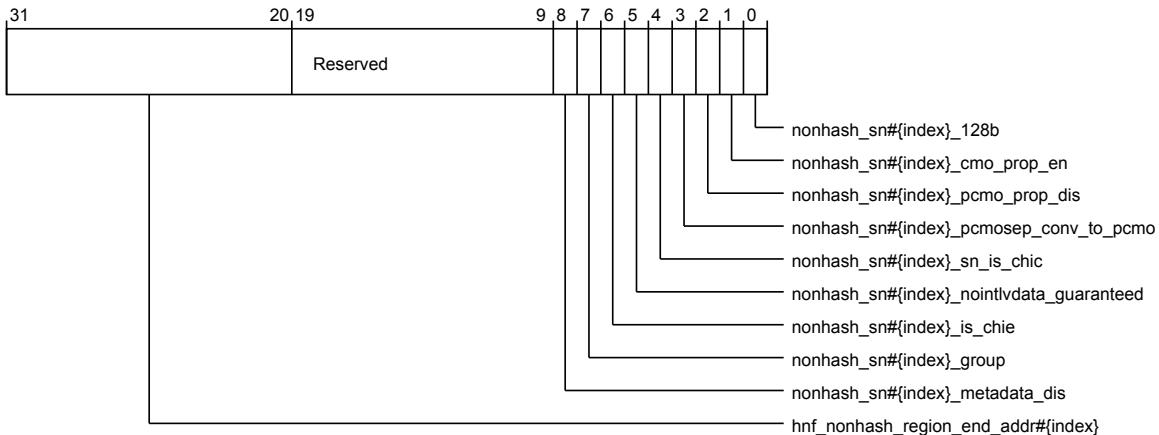


Figure 3-1407 por_hnf_por_hnf_sam_nonhash_cfg2_memregion_2-63 (low)

The following table shows the por_hnf_sam_nonhash_cfg2_memregion_2-63 lower register bit assignments.

Table 3-1427 por_hnf_por_hnf_sam_nonhash_cfg2_memregion_2-63 (low)

Bits	Field name	Description	Type	Reset
31:20	hnf_nonhash_region_end_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0
19:9	Reserved	Reserved	RO	-
8	nonhash_sn#{index}_metadata_dis	HNF implements metadata termination flow for nonhash SN #{index} when set	RW	1'b0
7	nonhash_sn#{index}_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
6	nonhash_sn#{index}_is_chie	nonhash SN #{index} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
5	nonhash_sn#{index}_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
4	nonhash_sn#{index}_sn_is_chic	Indicates that nonhash sn is a CHI-C SN when set	RW	1'b0
3	nonhash_sn#{index}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for nonhash SN #{index} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
2	nonhash_sn#{index}_pcmo_prop_dis	Disables PCMO propagation for nonhash SN #{index} when set	RW	1'b0

Table 3-1427 por_hnf_por_hnf_sam_nonhash_cfg2_memregion_2-63 (low) (continued)

Bits	Field name	Description	Type	Reset
1	nonhash_sn#{index}_cmo_prop_en	Enables CMO propagation for nonhash SN #{index} when set	RW	1'b0
0	nonhash_sn#{index}_128b	Data width of nonhash SN #{index} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0

por_hnf_sam_htg_cfg1_memregion_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Configures HTG memory region #{index} in HN-F SAM.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5400 + # {[0, 1, 2, 3, 4, 5, 6, 7]} * 8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

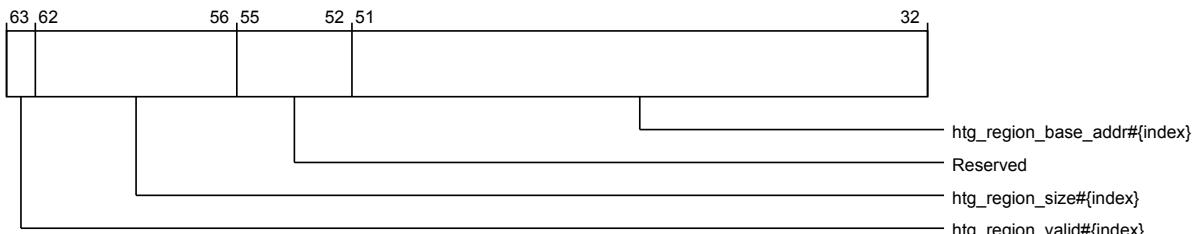


Figure 3-1408 por_hnf_por_hnf_sam_htg_cfg1_memregion_0-7 (high)

The following table shows the por_hnf_sam_htg_cfg1_memregion_0-7 higher register bit assignments.

Table 3-1428 por_hnf_por_hnf_sam_htg_cfg1_memregion_0-7 (high)

Bits	Field name	Description	Type	Reset
63	htg_region_valid#{index}	valid 1'b0: Not valid 1'b1: Valid for memory region comparison	RW	1'b0
62:56	htg_region_size#{index}	Memory region size CONSTRAINT: Memory region must be a power of two, from minimum size supported to maximum memory size ($2^{\text{address width}}$).	RW	7'b000000

Table 3-1428 por_hnf_por_hnf_sam_htg_cfg1_memregion_0-7 (high) (continued)

Bits	Field name	Description	Type	Reset
55:52	Reserved	Reserved	RO	-
51:32	htg_region_base_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0

The following image shows the lower register bit assignments.

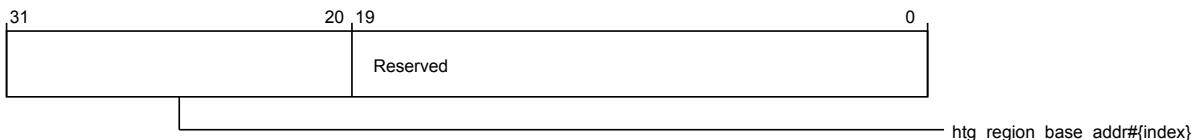


Figure 3-1409 por_hnf_por_hnf_sam_htg_cfg1_memregion_0-7 (low)

The following table shows the por_hnf_sam_htg_cfg1_memregion_0-7 lower register bit assignments.

Table 3-1429 por_hnf_por_hnf_sam_htg_cfg1_memregion_0-7 (low)

Bits	Field name	Description	Type	Reset
31:20	htg_region_base_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0
19:0	Reserved	Reserved	RO	-

por_hnf_sam_htg_cfg2_memregion_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Configures htg memory region #{index} in HN-F SAM.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5480 + # {[0, 1, 2, 3, 4, 5, 6, 7]} * 8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

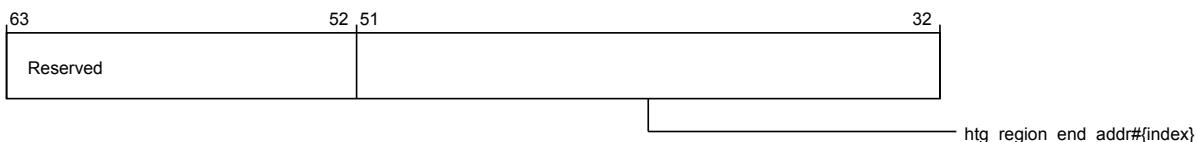


Figure 3-1410 por_hnf_por_hnf_sam_htg_cfg2_memregion_0-7 (high)

The following table shows the por_hnf_sam_htg_cfg2_memregion_0-7 higher register bit assignments.

Table 3-1430 por_hnf_por_hnf_sam_htg_cfg2_memregion_0-7 (high)

Bits	Field name	Description	Type	Reset
63:52	Reserved	Reserved	RO	-
51:32	htg_region_end_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0

The following image shows the lower register bit assignments.

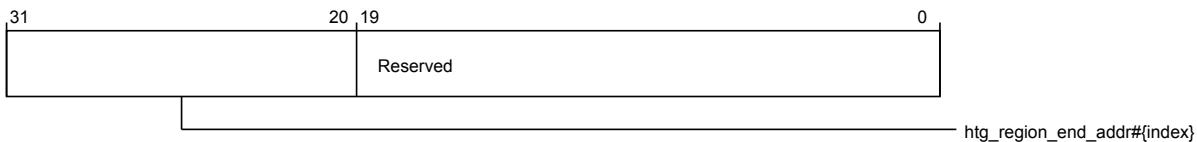


Figure 3-1411 por_hnf_por_hnf_sam_htg_cfg2_memregion_0-7 (low)

The following table shows the por_hnf_sam_htg_cfg2_memregion_0-7 lower register bit assignments.

Table 3-1431 por_hnf_por_hnf_sam_htg_cfg2_memregion_0-7 (low)

Bits	Field name	Description	Type	Reset
31:20	htg_region_end_addr#{index}	Bits [51:16] of base address of the range, LSB bit is defined by the parameter POR_HNSAM_RCOMP_LSB_PARAM	RW	32'h0
19:0	Reserved	Reserved	RO	-

por_hnf_sam_htg_cfg3_memregion_0-7

This register repeats 7 times. It parametrized by the index from 0 to 7. Configures the HTG memory region #{index}

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5500 + # {[0, 1, 2, 3, 4, 5, 6, 7]} * 8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

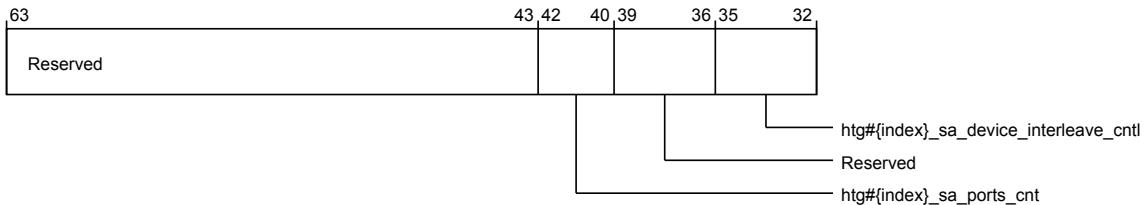


Figure 3-1412 por_hnf_por_hnf_sam_htg_cfg3_memregion_0-7 (high)

The following table shows the por_hnf_sam_htg_cfg3_memregion_0-7 higher register bit assignments.

Table 3-1432 por_hnf_por_hnf_sam_htg_cfg3_memregion_0-7 (high)

Bits	Field name	Description	Type	Reset
63:43	Reserved	Reserved	RO	-
42:40	htg#{index}_sa_ports_cnt	Specifies the number of CXSA/CXLSA device aggregated 3'b000: 1 port used 3'b001: 2 ports used 3'b010: 4 ports used 3'b011: 8 ports used 3'b100: 16 ports used others: Reserved	RW	3'b0
39:36	Reserved	Reserved	RO	-
35:32	htg#{index}_sa_device_interleave_cntl	This field controls the interleave size across all aggregated CXSA/CXLSA Devices 4'h0: 64B Interleaved 4'h1: 128B Interleaved 4'h2: 256B Interleaved 4'h3: 512B Interleaved .. 4'hF: 2MB Interleaved	RW	4'b0

The following image shows the lower register bit assignments.

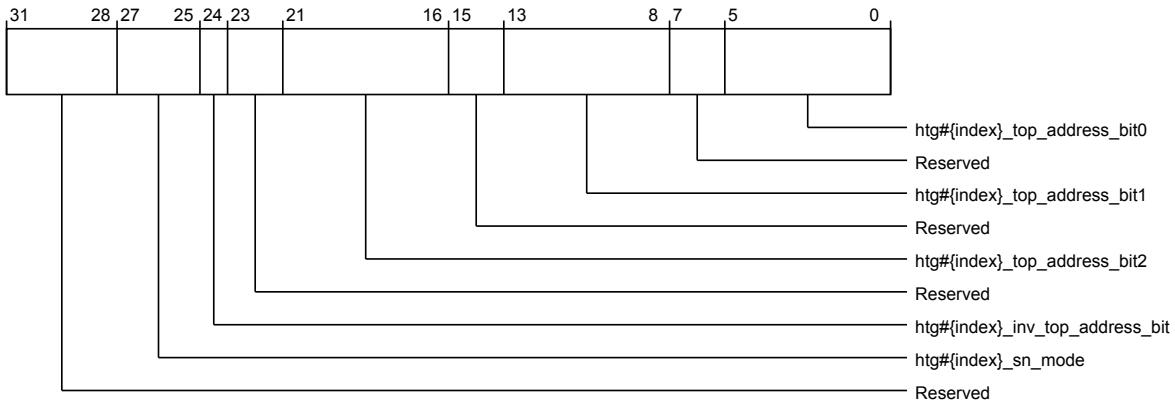


Figure 3-1413 por_hnf_por_hnf_sam_htg_cfg3_memregion_0-7 (low)

The following table shows the por_hnf_sam_htg_cfg3_memregion_0-7 lower register bit assignments.

Table 3-1433 por_hnf_por_hnf_sam_htg_cfg3_memregion_0-7 (low)

Bits	Field name	Description	Type	Reset
31:28	Reserved	Reserved	RO	-
27:25	htg#{index}_sn_mode	SN selection mode 3'b000: Reserved 3'b001: 3-SN mode (SN0, SN1, SN2) 3'b010: 6-SN mode (SN0, SN1, SN2, SN3, SN4, SN5) 3'b011: 5-SN mode (SN0, SN1, SN2, SN3, SN4) 3'b100: 2-SN mode (SN0, SN1) power of 2 hashing 3'b101: 4-SN mode (SN0, SN1, SN2, SN3) power of 2 hashing 3'b110: 8-SN mode (SN0, SN1, SN2, SN3, SN4, SN5, SN6, SN7) power of 2 hashing 3'b111: CXSA/CXLSA aggregated SA selection function	RW	3'b0
24	htg#{index}_inv_top_address_bit	Inverts the top address bit (top_address_bit1 if 3-SN, top_address_bit2 if 6-SN); only used when the address map does not have unique address bit combinations	RW	1'h0
23:22	Reserved	Reserved	RO	-
21:16	htg#{index}_top_address_bit2	Top address bit 2	RW	6'h00
15:14	Reserved	Reserved	RO	-
13:8	htg#{index}_top_address_bit1	Top address bit 1	RW	6'h00
7:6	Reserved	Reserved	RO	-
5:0	htg#{index}_top_address_bit0	Top address bit 0	RW	6'h00

por_hnf_sam_htg_sn_nodeid_reg_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Configures SN node IDs for HTGs in the HNSAM . Controls target SN node IDs # $\{index*4 + 0\}$ to # $\{index*4 + 3\}$.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5600 + # $\{[0, 1, 2, \dots, 14, 15]\} * 8$
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

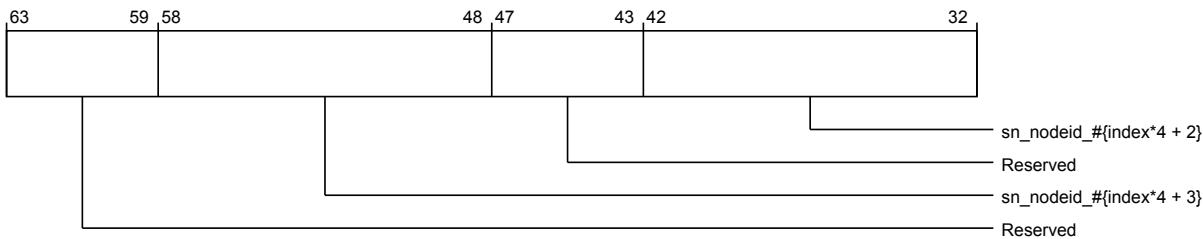


Figure 3-1414 por_hnf_por_hnf_sam_htg_sn_nodeid_reg_0-15 (high)

The following table shows the por_hnf_sam_htg_sn_nodeid_reg_0-15 higher register bit assignments.

Table 3-1434 por_hnf_por_hnf_sam_htg_sn_nodeid_reg_0-15 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	sn_nodeid_{index*4 + 3}	Hashed target SN node ID # $\{index*4 + 3\}$	RW	11'b000000000000
47:43	Reserved	Reserved	RO	-
42:32	sn_nodeid_{index*4 + 2}	Hashed target SN node ID # $\{index*4 + 2\}$	RW	11'b000000000000

The following image shows the lower register bit assignments.

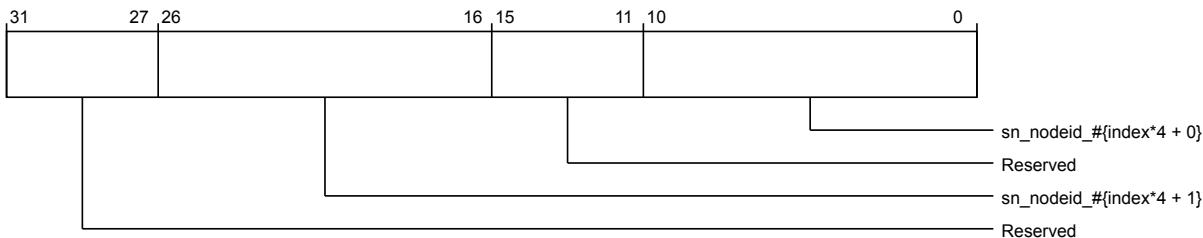


Figure 3-1415 por_hnf_por_hnf_sam_htg_sn_nodeid_reg_0-15 (low)

The following table shows the por_hnf_sam_htg_sn_nodeid_reg_0-15 lower register bit assignments.

Table 3-1435 por_hnf_por_hnf_sam_htg_sn_nodeid_reg_0-15 (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:16	sn_nodeid_{index*4 + 1}	Hashed target SN node ID #{index*4 + 1}	RW	11'b000000000000
15:11	Reserved	Reserved	RO	-
10:0	sn_nodeid_{index*4 + 0}	Hashed target SN node ID #{index*4 + 0}	RW	11'b000000000000

por_hnf_sam_htg_sn_attr_0-15

This register repeats 15 times. It parametrized by the index from 0 to 15. Configures SN node attributes HTGs in the HNSAM . Controls SN attributes #{index*4 + 0} to #{index*4 + 3}.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5680 + {[0, 1, 2, .., 14, 15]}*8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

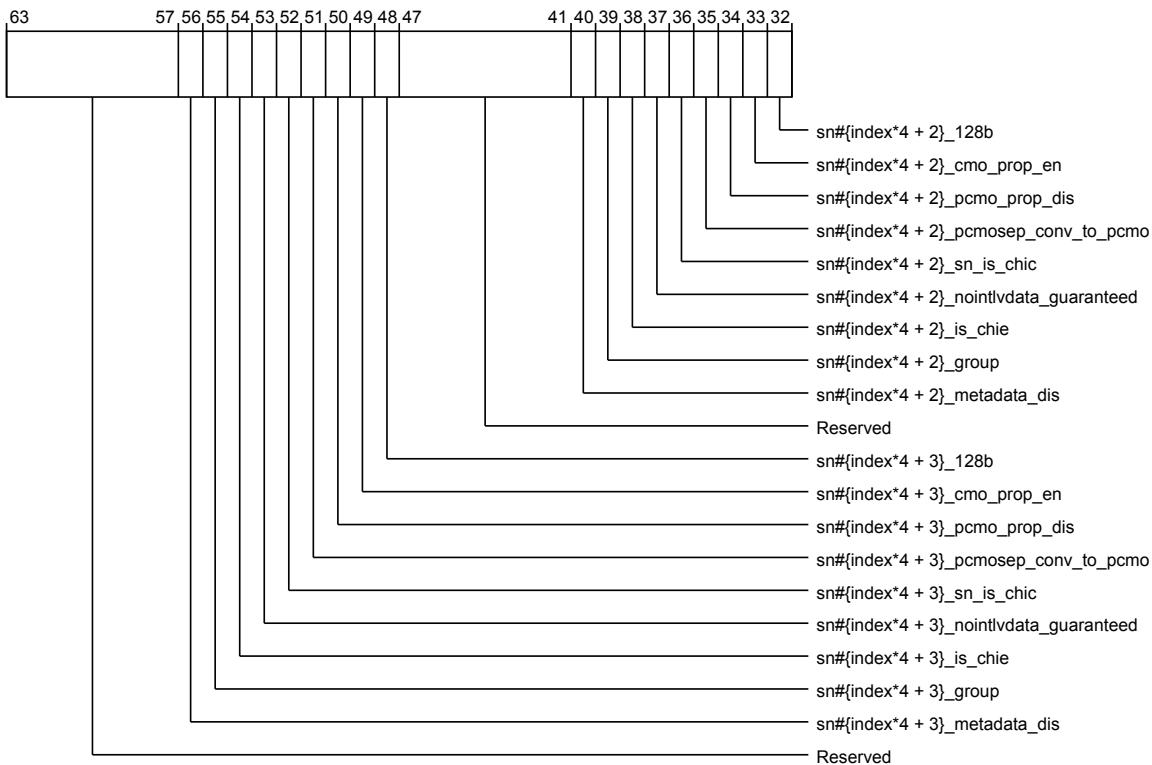


Figure 3-1416 por_hnf_por_hnf_sam_htg_sn_attr_0-15 (high)

The following table shows the por_hnf_sam_htg_sn_attr_0-15 higher register bit assignments.

Table 3-1436 por_hnf_por_hnf_sam_htg_sn_attr_0-15 (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56	sn#{index*4 + 3}_metadata_dis	HNF implements metadata termination flow for SN #{index*4 + 3} when set	RW	1'b0
55	sn#{index*4 + 3}_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
54	sn#{index*4 + 3}_is_chie	SN #{index*4 + 3} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
53	sn#{index*4 + 3}_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
52	sn#{index*4 + 3}_sn_is_chic	Indicates that sn is a CHI-C SN when set	RW	1'b0
51	sn#{index*4 + 3}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN #{index*4 + 3} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
50	sn#{index*4 + 3}_pcmoprop_dis	Disables PCMO propagation for SN #{index*4 + 3} when set	RW	1'b0
49	sn#{index*4 + 3}_cmo_prop_en	Enables CMO propagation for SN #{index*4 + 3} when set	RW	1'b0
48	sn#{index*4 + 3}_128b	Data width of SN #{index*4 + 3} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
47:41	Reserved	Reserved	RO	-
40	sn#{index*4 + 2}_metadata_dis	HNF implements metadata termination flow for SN #{index*4 + 2} when set	RW	1'b0
39	sn#{index*4 + 2}_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
38	sn#{index*4 + 2}_is_chie	SN #{index*4 + 2} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
37	sn#{index*4 + 2}_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
36	sn#{index*4 + 2}_sn_is_chic	Indicates that sn is a CHI-C SN when set	RW	1'b0

Table 3-1436 por_hnf_por_hnf_sam_htg_sn_attr_0-15 (high) (continued)

Bits	Field name	Description	Type	Reset
35	sn#{index*4 + 2}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN #{index*4 + 2} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
34	sn#{index*4 + 2}_pcm_o_prop_dis	Disables PCMO propagation for SN #{index*4 + 2} when set	RW	1'b0
33	sn#{index*4 + 2}_cmo_prop_en	Enables CMO propagation for SN #{index*4 + 2} when set	RW	1'b0
32	sn#{index*4 + 2}_128b	Data width of SN #{index*4 + 2} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0

The following image shows the lower register bit assignments.

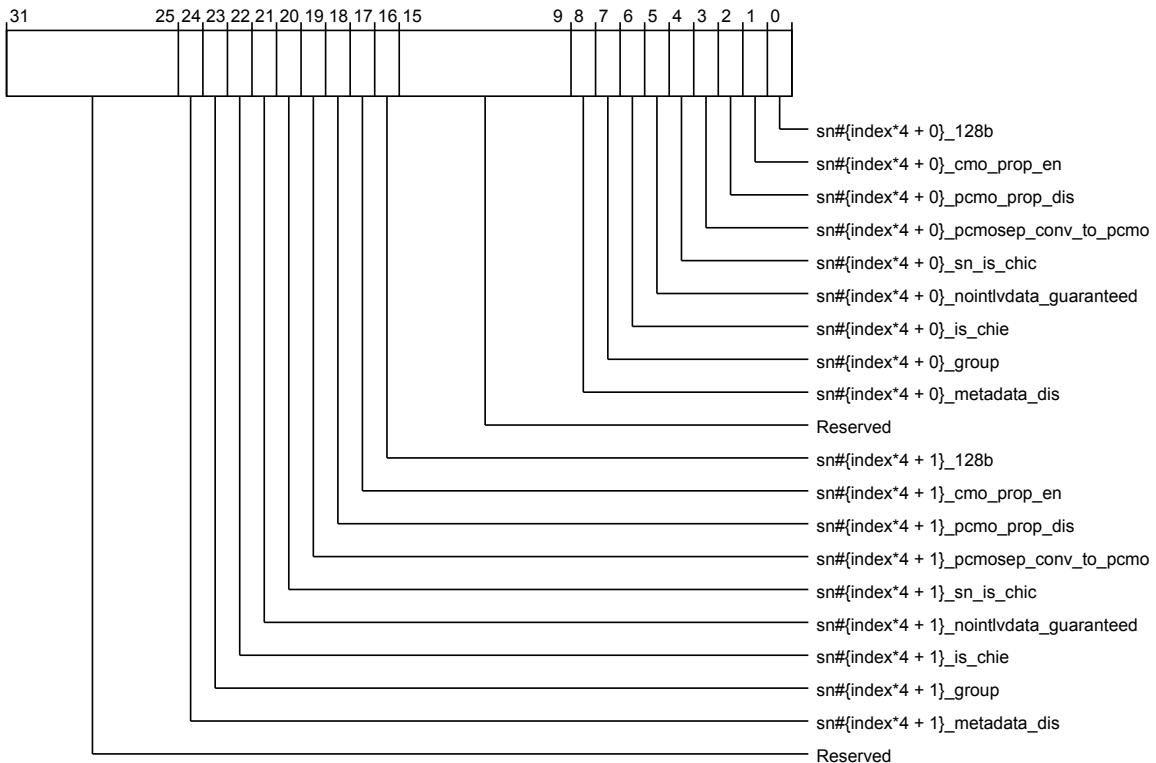


Figure 3-1417 por_hnf_por_hnf_sam_htg_sn_attr_0-15 (low)

The following table shows the por_hnf_sam_htg_sn_attr_0-15 lower register bit assignments.

Table 3-1437 por_hnf_por_hnf_sam_htg_sn_attr_0-15 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	sn#{index*4 + 1}_metadata_dis	HNF implements metadata termination flow for SN #{index*4 + 1} when set	RW	1'b0
23	sn#{index*4 + 1}_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
22	sn#{index*4 + 1}_is_chie	SN #{index*4 + 1} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
21	sn#{index*4 + 1}_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
20	sn#{index*4 + 1}_sn_is_chic	Indicates that sn is a CHI-C SN when set	RW	1'b0
19	sn#{index*4 + 1}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN #{index*4 + 1} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
18	sn#{index*4 + 1}_pcmoprop_dis	Disables PCMO propagation for SN #{index*4 + 1} when set	RW	1'b0
17	sn#{index*4 + 1}_cmo_prop_en	Enables CMO propagation for SN #{index*4 + 1} when set	RW	1'b0
16	sn#{index*4 + 1}_128b	Data width of SN #{index*4 + 1} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
15:9	Reserved	Reserved	RO	-
8	sn#{index*4 + 0}_metadata_dis	HNF implements metadata termination flow for SN #{index*4 + 0} when set	RW	1'b0
7	sn#{index*4 + 0}_group	Specifies the SN-F grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
6	sn#{index*4 + 0}_is_chie	SN #{index*4 + 0} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
5	sn#{index*4 + 0}_nointlvdata_guaranteed	SN guarantees the return data will not be interleaved	RW	1'b0
4	sn#{index*4 + 0}_sn_is_chic	Indicates that sn is a CHI-C SN when set	RW	1'b0

Table 3-1437 por_hnf_por_hnf_sam_htg_sn_attr_0-15 (low) (continued)

Bits	Field name	Description	Type	Reset
3	sn#{index*4 + 0}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for SN #{index*4 + 0} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
2	sn#{index*4 + 0}_pcmo_prop_dis	Disables PCMO propagation for SN #{index*4 + 0} when set	RW	1'b0
1	sn#{index*4 + 0}_cmo_prop_en	Enables CMO propagation for SN #{index*4 + 0} when set	RW	1'b0
0	sn#{index*4 + 0}_128b	Data width of SN #{index*4 + 0} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0

por_hnf_sam_ccg_sa_nodeid_reg_0-3

This register repeats 3 times. It parametrized by the index from 0 to 3. Configures CCG SA node IDs for HTGs in the HNSAM

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h5700 + # {[0, 1,

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

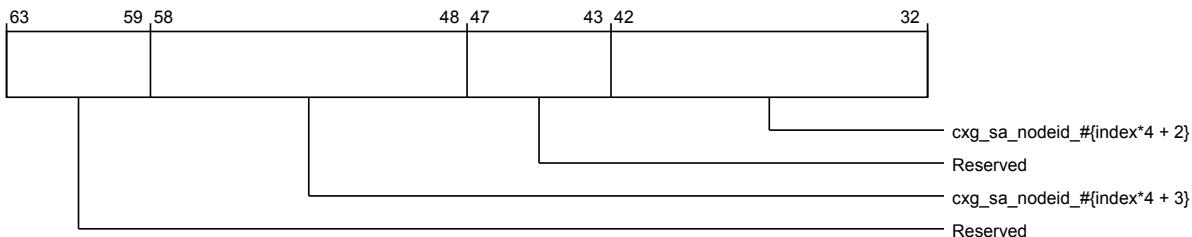


Figure 3-1418 por_hnf_por_hnf_sam_ccg_sa_nodeid_reg_0-3 (high)

The following table shows the port assignments for the higher register bit assignments.

Table 3-1438 por_hnf_por_hnf_sam_ccg_sa_nodeid_reg_0-3 (high)

Bits	Field name	Description	Type	Reset
63:59	Reserved	Reserved	RO	-
58:48	cxg_sa_nodeid_{index*4 + 3}	Hashed target CCG SA node ID #{index*4 + 3}	RW	11'b000000000000

Table 3-1438 por_hnf_por_hnf_sam_ccg_sa_nodeid_reg_0-3 (high) (continued)

Bits	Field name	Description	Type	Reset
47:43	Reserved	Reserved	RO	-
42:32	cxg_sa_nodeid_{index*4 + 2}	Hashed target CCG SA node ID #{index*4 + 2}	RW	11'b000000000000

The following image shows the lower register bit assignments.

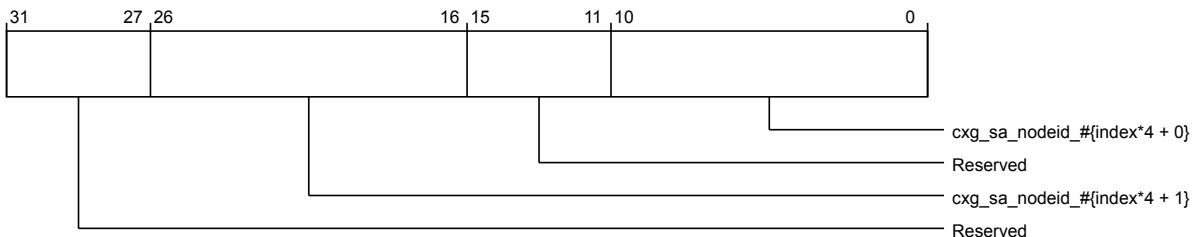


Figure 3-1419 por_hnf_por_hnf_sam_ccg_sa_nodeid_reg_0-3 (low)

The following table shows the por_hnf_sam_ccg_sa_nodeid_reg_0-3 lower register bit assignments.

Table 3-1439 por_hnf_por_hnf_sam_ccg_sa_nodeid_reg_0-3 (low)

Bits	Field name	Description	Type	Reset
31:27	Reserved	Reserved	RO	-
26:16	cxg_sa_nodeid_{index*4 + 1}	Hashed target CCG SA node ID #{index*4 + 1}	RW	11'b000000000000
15:11	Reserved	Reserved	RO	-
10:0	cxg_sa_nodeid_{index*4 + 0}	Hashed target CCG SA node ID #{index*4 + 0}	RW	11'b000000000000

por_hnf_sam_ccg_sa_attr_0-3

This register repeats 3 times. It parametrized by the index from 0 to 3. Configures CCG SA node attributes.

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5740 + {[0, 1, 2, 3]}*8
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

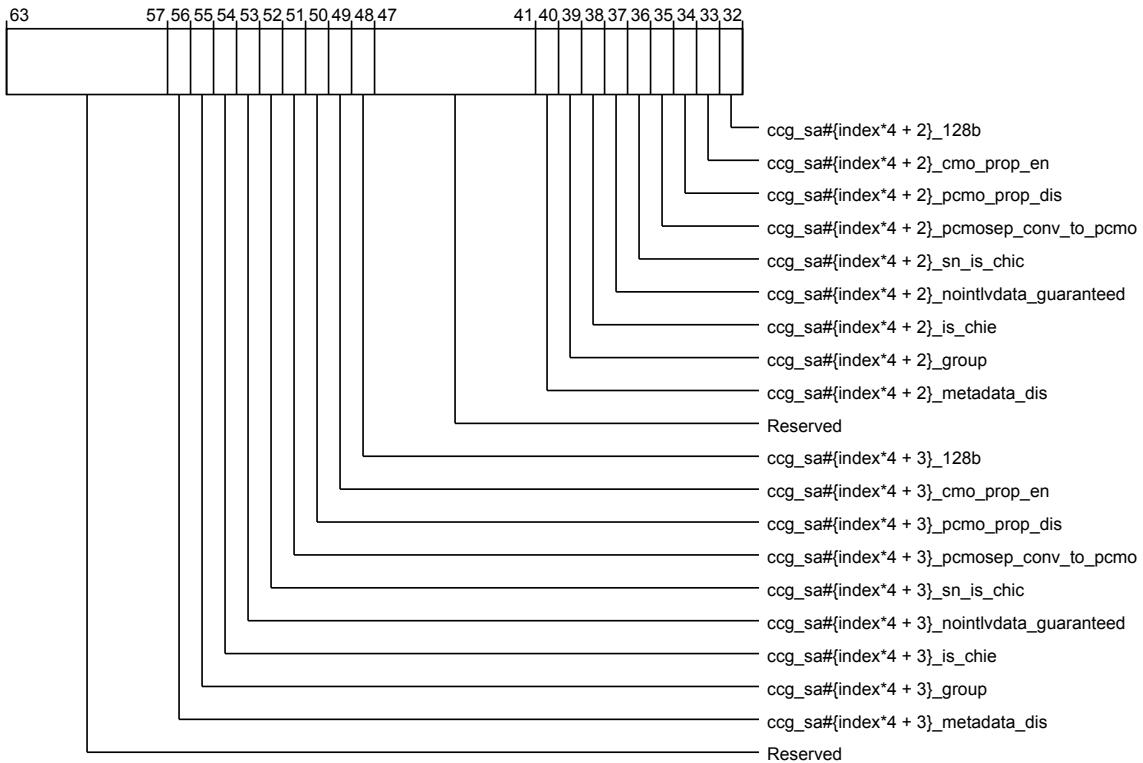


Figure 3-1420 por_hnf_por_hnf_sam_ccg_sa_attr_0-3 (high)

The following table shows the por_hnf_sam_ccg_sa_attr_0-3 higher register bit assignments.

Table 3-1440 por_hnf_por_hnf_sam_ccg_sa_attr_0-3 (high)

Bits	Field name	Description	Type	Reset
63:57	Reserved	Reserved	RO	-
56	ccg_sa#{index*4 + 3}_metadata_dis	HNF implements metadata termination flow for CCG_SA #{index*4 + 3} when set	RW	1'b0
55	ccg_sa#{index*4 + 3}_group	Specifies the CCG_SA grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
54	ccg_sa#{index*4 + 3}_is_chie	CCG_SA #{index*4 + 3} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
53	ccg_sa#{index*4 + 3}_nointlvdata_guaranteed	CCG_SA guarantees the return data will not be interleaved	RW	1'b0
52	ccg_sa#{index*4 + 3}_sn_is_chic	Indicates that CCG_SA is a CHI-C CCG_SA when set	RW	1'b0
51	ccg_sa#{index*4 + 3}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for CCG_SA #{index*4 + 3} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
50	ccg_sa#{index*4 + 3}_pcmo_prop_dis	Disables PCMO propagation for CCG_SA #{index*4 + 3} when set	RW	1'b0

Table 3-1440 por_hnf_por_hnf_sam_ccg_sa_attr_0-3 (high) (continued)

Bits	Field name	Description	Type	Reset
49	ccg_sa#{index*4 + 3}_cmo_prop_en	Enables CMO propagation for CCG_SA #{index*4 + 3} when set	RW	1'b0
48	ccg_sa#{index*4 + 3}_128b	Data width of CCG_SA #{index*4 + 3} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
47:41	Reserved	Reserved	RO	-
40	ccg_sa#{index*4 + 2}_metadata_dis	HNF implements metadata termination flow for CCG_SA #{index*4 + 2} when set	RW	1'b0
39	ccg_sa#{index*4 + 2}_group	Specifies the CCG_SA grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
38	ccg_sa#{index*4 + 2}_is_chie	CCG_SA #{index*4 + 2} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
37	ccg_sa#{index*4 + 2}_nointlvdata_guaranteed	CCG_SA guarantees the return data will not be interleaved	RW	1'b0
36	ccg_sa#{index*4 + 2}_sn_is_chic	Indicates that CCG_SA is a CHI-C CCG_SA when set	RW	1'b0
35	ccg_sa#{index*4 + 2}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for CCG_SA #{index*4 + 2} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
34	ccg_sa#{index*4 + 2}_pcmo_prop_dis	Disables PCMO propagation for CCG_SA #{index*4 + 2} when set	RW	1'b0
33	ccg_sa#{index*4 + 2}_cmo_prop_en	Enables CMO propagation for CCG_SA #{index*4 + 2} when set	RW	1'b0
32	ccg_sa#{index*4 + 2}_128b	Data width of CCG_SA #{index*4 + 2} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0

The following image shows the lower register bit assignments.

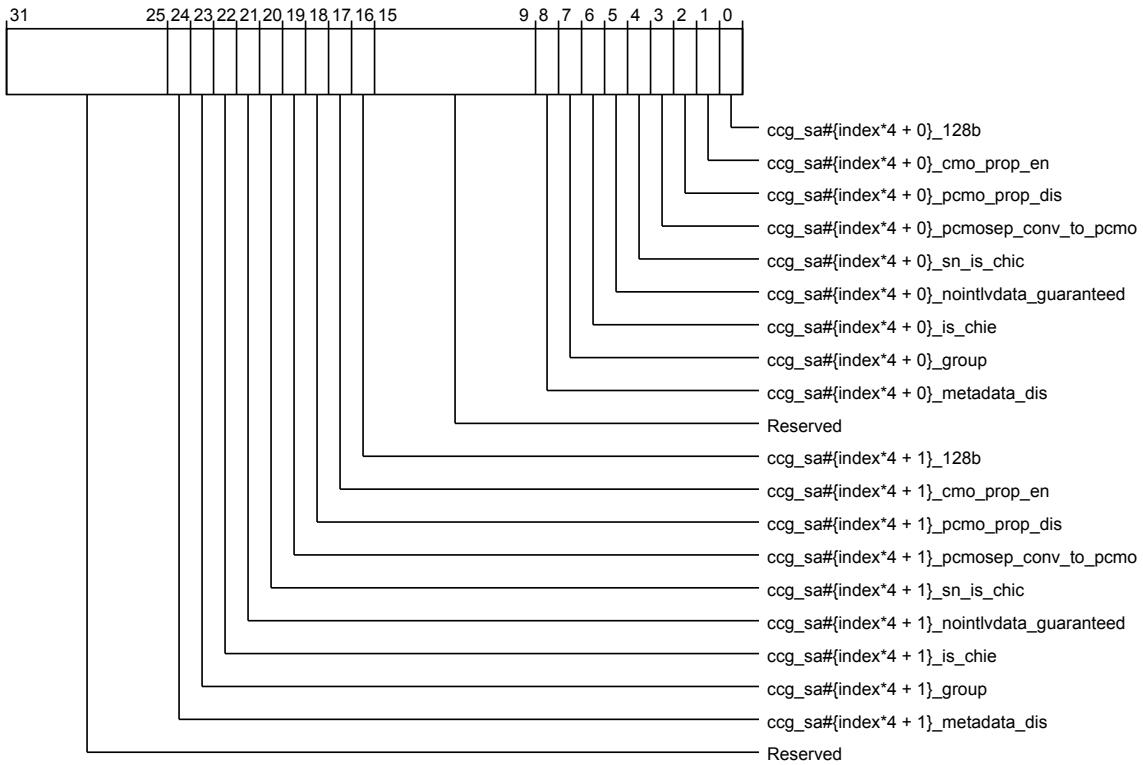


Figure 3-1421 por_hnf_por_hnf_sam_ccg_sa_attr_0-3 (low)

The following table shows the por_hnf_sam_ccg_sa_attr_0-3 lower register bit assignments.

Table 3-1441 por_hnf_por_hnf_sam_ccg_sa_attr_0-3 (low)

Bits	Field name	Description	Type	Reset
31:25	Reserved	Reserved	RO	-
24	ccg_sa#{index*4 + 1}_metadata_dis	HNF implements metadata termination flow for CCG_SA #{index*4 + 1} when set	RW	1'b0
23	ccg_sa#{index*4 + 1}_group	Specifies the CCG_SA grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
22	ccg_sa#{index*4 + 1}_is_chie	CCG_SA #{index*4 + 1} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
21	ccg_sa#{index*4 + 1}_nointlvdata_guaranteed	CCG_SA guarantees the return data will not be interleaved	RW	1'b0
20	ccg_sa#{index*4 + 1}_sn_is_chic	Indicates that CCG_SA is a CHI-C CCG_SA when set	RW	1'b0
19	ccg_sa#{index*4 + 1}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for CCG_SA #{index*4 + 1} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
18	ccg_sa#{index*4 + 1}_pcmo_prop_dis	Disables PCMO propagation for CCG_SA #{index*4 + 1} when set	RW	1'b0

Table 3-1441 por_hnf_por_hnf_sam_ccg_sa_attr_0-3 (low) (continued)

Bits	Field name	Description	Type	Reset
17	ccg_sa#{index*4 + 1}_cmo_prop_en	Enables CMO propagation for CCG_SA #{index*4 + 1} when set	RW	1'b0
16	ccg_sa#{index*4 + 1}_128b	Data width of CCG_SA #{index*4 + 1} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0
15:9	Reserved	Reserved	RO	-
8	ccg_sa#{index*4 + 0}_metadata_dis	HNF implements metadata termination flow for CCG_SA #{index*4 + 0} when set	RW	1'b0
7	ccg_sa#{index*4 + 0}_group	Specifies the CCG_SA grouping 1'b0: Group A 1'b1: Group B	RW	1'b0
6	ccg_sa#{index*4 + 0}_is_chie	CCG_SA #{index*4 + 0} supports CHI-E (Not applicable in CMN-600 Rhodes)	RW	1'b0
5	ccg_sa#{index*4 + 0}_nointlvdata_guaranteed	CCG_SA guarantees the return data will not be interleaved	RW	1'b0
4	ccg_sa#{index*4 + 0}_sn_is_chic	Indicates that CCG_SA is a CHI-C CCG_SA when set	RW	1'b0
3	ccg_sa#{index*4 + 0}_pcmosep_conv_to_pcmo	Convert CleanSharedPersistSep to CleanSharedPersist for CCG_SA #{index*4 + 0} when set CONSTRAINT: Should not be enabled when sn_pcmo_prop_dis bit is set to 1	RW	1'b0
2	ccg_sa#{index*4 + 0}_pcmo_prop_dis	Disables PCMO propagation for CCG_SA #{index*4 + 0} when set	RW	1'b0
1	ccg_sa#{index*4 + 0}_cmo_prop_en	Enables CMO propagation for CCG_SA #{index*4 + 0} when set	RW	1'b0
0	ccg_sa#{index*4 + 0}_128b	Data width of CCG_SA #{index*4 + 0} 1'b1: 128 bits 1'b0: 256 bits	RW	1'b0

hnf_generic_regs_0-1

This register repeats 1 times. It parametrized by the index from 0 to 1. Configuration register for the custom logic

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h5780 + # {[0, 1]} * 8

Register reset 64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hnf_secure_register_groups_override.cfg_ctl

The following image shows the higher register bit assignments.

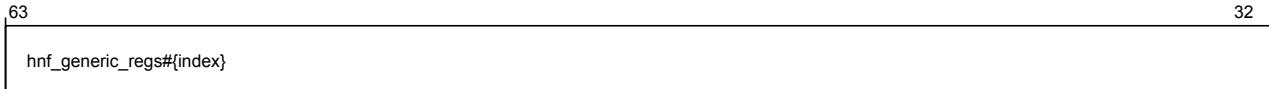


Figure 3-1422 por_hnf_hnf_generic_regs_0-1 (high)

The following table shows the hnf_generic_regs_0-1 higher register bit assignments.

Table 3-1442 por_hnf_hnf_generic_regs_0-1 (high)

Bits	Field name	Description	Type	Reset
63:32	hnf_generic_regs#{index}	Configuration register for the custom logic	RW	64'h0

The following image shows the lower register bit assignments.

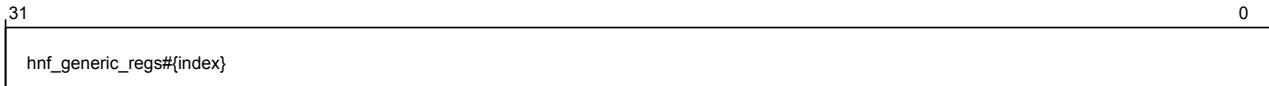


Figure 3-1423 por_hnf_hnf_generic_regs_0-1 (low)

The following table shows the hnf_generic_regs_0-1 lower register bit assignments.

Table 3-1443 por_hnf_hnf_generic_regs_0-1 (low)

Bits	Field name	Description	Type	Reset
31:0	hnf_generic_regs#{index}	Configuration register for the custom logic	RW	64'h0

por_hnf_pa2setaddr_slc

Functions as the control register of PA to SetAddr and vice versa conversion for HNF-SLC

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h5900

Register reset 64'b0110

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hnf_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.

63

32

Reserved

Figure 3-1424 por_hnf_por_hnf_pa2setaddr_slc (high)

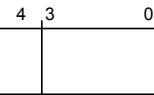
The following table shows the por_hnf_pa2setaddr_slc higher register bit assignments.

Table 3-1444 por_hnf_por_hnf_pa2setaddr_slc (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

31



setaddr_startbit_slc

Figure 3-1425 por_hnf_por_hnf_pa2setaddr_slc (low)

The following table shows the por_hnf_pa2setaddr_slc lower register bit assignments.

Table 3-1445 por_hnf_por_hnf_pa2setaddr_slc (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	setaddr_startbit_slc	SLC: SetAddr starting bit for SLC TODO add a description here abt contiguous bits 4'b0110: Setaddr starts from PA[6] 4'b0111: Setaddr starts from PA[7] 4'b1000: Setaddr starts from PA[8] 4'b1001: Setaddr starts from PA[9] 4'b1010: Setaddr starts from PA[10] 4'b1011: Setaddr starts from PA[11] 4'b1100: Setaddr starts from PA[12]	RW	4'b0110

por_hnf_pa2setaddr_sf

Functions as the control register of PA to Set/TagAddr and vice versa conversion for HNF-SF

Its characteristics are:

Type RW

Register width (Bits) 64

Address offset 16'h5908

Register reset	64'b0110
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.

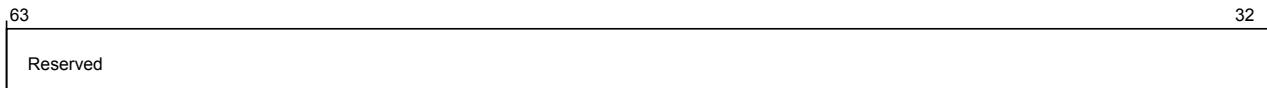


Figure 3-1426 por_hnf_por_hnf_pa2setaddr_sf (high)

The following table shows the por_hnf_pa2setaddr_sf higher register bit assignments.

Table 3-1446 por_hnf_por_hnf_pa2setaddr_sf (high)

Bits	Field name	Description	Type	Reset
63:32	Reserved	Reserved	RO	-

The following image shows the lower register bit assignments.

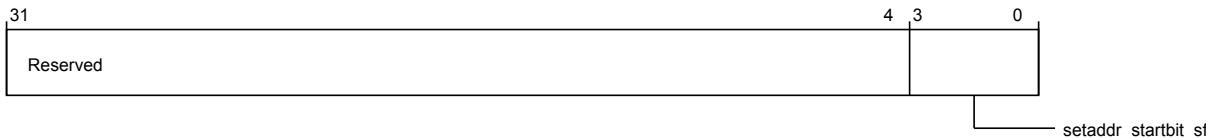


Figure 3-1427 por_hnf_por_hnf_pa2setaddr_sf (low)

The following table shows the por_hnf_pa2setaddr_sf lower register bit assignments.

Table 3-1447 por_hnf_por_hnf_pa2setaddr_sf (low)

Bits	Field name	Description	Type	Reset
31:4	Reserved	Reserved	RO	-
3:0	setaddr_startbit_sf	SF: SetAddr starting bit for SF 4'b0110: Setaddr starts from PA[6] 4'b0111: Setaddr starts from PA[7] 4'b1000: Setaddr starts from PA[8] 4'b1001: Setaddr starts from PA[9] 4'b1010: Setaddr starts from PA[10] 4'b1011: Setaddr starts from PA[11] 4'b1100: Setaddr starts from PA[12]	RW	4'b0110

por_hnf_pa2setaddr_flex_slc

Functions as the SLC control register of PA to Set/TagAddr and vice versa conversion for HNF (flexible)

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5910
Register reset	64'b0
Usage constraints	Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.
Secure group override	por_hnf_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.

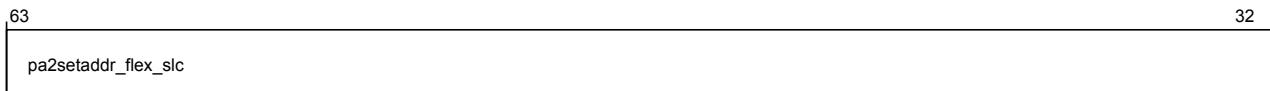


Figure 3-1428 por_hnf_por_hnf_pa2setaddr flex_slc (high)

The following table shows the por_hnf_pa2setaddr flex_slc higher register bit assignments.

Table 3-1448 por_hnf_por_hnf_pa2setaddr flex_slc (high)

Bits	Field name	Description	Type	Reset
63:32	pa2setaddr flex_slc	FLEXIBLE: PA to SET/TAG ADDR and vice versa conversion config field for SLC	RW	64'b0

The following image shows the lower register bit assignments.



Figure 3-1429 por_hnf_por_hnf_pa2setaddr flex_slc (low)

The following table shows the por_hnf_pa2setaddr flex_slc lower register bit assignments.

Table 3-1449 por_hnf_por_hnf_pa2setaddr flex_slc (low)

Bits	Field name	Description	Type	Reset
31:0	pa2setaddr flex_slc	FLEXIBLE: PA to SET/TAG ADDR and vice versa conversion config field for SLC	RW	64'b0

por_hnf_pa2setaddr flex_sf

Functions as the SF control register of PA to Set/TagAddr and vice versa conversion for HNF (flexible)

Its characteristics are:

Type	RW
Register width (Bits)	64
Address offset	16'h5918
Register reset	64'b0

Usage constraints Only accessible by secure accesses. Writes to this register must occur prior to the first non-configuration access targeting the device.

Secure group override por_hnf_secure_register_groups_override.pa2setaddr_ctl

The following image shows the higher register bit assignments.

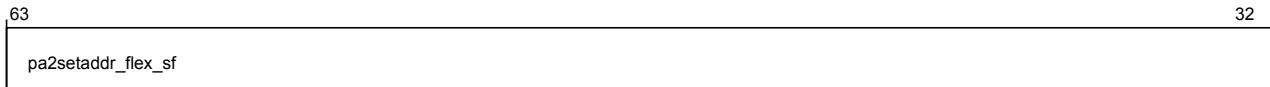


Figure 3-1430 por_hnf_por_hnf_pa2setaddr flex_sf (high)

The following table shows the por_hnf_pa2setaddr_flex_sf higher register bit assignments.

Table 3-1450 por_hnf_por_hnf_pa2setaddr flex_sf (high)

Bits	Field name	Description	Type	Reset
63:32	pa2setaddr_flex_sf	FLEXIBLE: PA to SET/TAG ADDR conversion and vice versa config field for SF	RW	64'b0

The following image shows the lower register bit assignments.

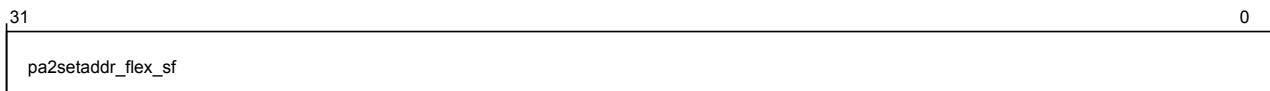


Figure 3-1431 por_hnf_por_hnf_pa2setaddr flex_sf (low)

The following table shows the por_hnf_pa2setaddr_flex_sf lower register bit assignments.

Table 3-1451 por_hnf_por_hnf_pa2setaddr flex_sf (low)

Bits	Field name	Description	Type	Reset
31:0	pa2setaddr_flex_sf	FLEXIBLE: PA to SET/TAG ADDR conversion and vice versa config field for SF	RW	64'b0

3.4 CMN-700 programming

This section contains CMN-700 programming information.

This section contains the following subsections:

- [3.4.1 Boot-time programming sequence on page 3-1238](#).
- [3.4.2 Run-time programming requirements on page 3-1239](#).
- [3.4.3 RN SAM and HN-F SAM programming on page 3-1239](#).
- [3.4.4 Program the dual DAT/RSP channel selection scheme on page 3-1243](#).
- [3.4.5 Program non-XY routing registers on page 3-1243](#).
- [3.4.6 RN-I and HN-I PCIe programming sequence on page 3-1244](#).
- [3.4.7 MTSX programming on page 3-1245](#).
- [3.4.8 DT programming on page 3-1253](#).
- [3.4.9 PMU system programming on page 3-1254](#).

3.4.1 Boot-time programming sequence

A specific boot-time programming sequence must be used to set up CMN-700 correctly. An example sequence is provided, which uses a *System Control Processor* (SCP) to perform the initial boot configuration.

After reset, the following configuration steps must happen before broad access to CMN-700 components is available:

1. CMN-700 uses a default configuration to access boot flash through the HN-D ACE-Lite master interface and also the configuration registers.
2. An RN-F, or a master that is connected to an RN-I, must then access the configuration registers to configure CMN-700. This boot-time configuration must happen before there is broader access to components such as HN-F or SN.

The following example provides more information on the boot process. It assumes an SCP is performing the CMN-700 configuration.

1. The SCP boots, either from local memory or through CMN-700 memory accesses targeting memory behind the HN-D:
 - All other masters are either held in reset or issue no requests to CMN-700 until the boot programming is complete.
 - The HN-D is identified through straps on the RN SAM.
2. If necessary, the SCP discovers the system.
3. The SCP determines the wanted address map and corresponding SAM register values.
4. If necessary, the SCP remaps the configuration register space by completing the following steps:
 - a. It drains all requests in flight by waiting for their responses.
 - b. It issues a single 64-bit store to a PERIPHBASE register behind the HN-D. This register would be in logic that is external to CMN-700 and an update would cause the signal values on the CFGM_PERIPHBASE input to change.
 - c. It waits for the response for that store.
5. If necessary, the SCP writes to the CMN-700 configuration registers to program the SAM for all HN-Fs.
6. The SCP writes to the CMN-700 configuration registers to program the SAM for all RNs including the one being used by the SCP.

Note

RN-F ESAM types are not able to block transactions before RN SAM programming and some external mechanism must block transactions. CMN-700 only supports RN-F ESAM types.

After programming the SAM for all RNs, the SCP sets a bit that enables use of the programmed address map instead of the default address map. This bit indicates that the SAM setup is complete.

Once the preceding steps are complete, the SCP can make general accesses anywhere in the address space and other masters can begin issuing requests.

3.4.2 Run-time programming requirements

This section describes the requirements for programming during runtime.

The hardware for handling RN membership in the coherence domain or DVM domain has been shifted to the XP to which the RN is attached. A low-level four-phase handshake mechanism (**SYSCOREQ**/**SYSCOACK**) has been added to allow quick and local entry to and exit from snoop and DVM domains. No communication with central hardware resources is needed. When a block is removed from the coherence or DVM domain, the XP acts as a protocol agent to give a generic response to any snoop or DVM messages.

For legacy devices that do not support the **SYSCOREQ/SYSCOACK** mechanism, direct configuration writes to the XP by software can trigger the same mechanism. For more information, see [2.3.9 RN entry to and exit from Snoop and DVM domains on page 2-92](#).

3.4.3 RN SAM and HN-F SAM programming

You must follow specific programming sequences to set up the RN SAM and HN-F SAM correctly. The register operating modes and encodings you use depend on your system configuration and requirements.

Program the SAM

The SAM must be programmed using a specific sequence. An RN-F or master that is connected to an RN-I must perform this sequence during the configuration of CMN-700 at boot.

There are several configuration decisions that must be made when setting up the SAM. For more information, see the following sections:

- [2.4.5 RN SAM on page 2-105](#)
- [2.4.7 HN-F SAM on page 2-125](#)

Prerequisites

This sequence is part of the overall CMN-700 boot configuration process. There are steps that must occur at boot-time before SAM programming. For more information about the full process, see [3.4.1 Boot-time programming sequence on page 3-1238](#).

All **MBISTREQ** and **nMBISTRESET** signals must be disabled during functional operation. The P-Channel, Q-Channel, **ACLKEN_M**, and **ACLKEN_S** signals must also all be set correctly for SAM programming.

Procedure

1. Define the following memory map regions:
 - Hashed memory regions, which target HN-Fs. The hashed memory regions can be partitioned into SCGs, if applicable.
 - Non-hashed memory regions, which likely target HN-I or HN-D.
 - Non-hashed memory regions, which likely target HN-I, HN-D, or HN-P.
 - Non-hashed regions with HN-I, HN-D, or HN-P mapping. If a single HN-F is the target, HN-F can also be used in non-hashed mode.
 - GIC memory region, if present.
 - HN-F SAM memory regions, if applicable.
 - Mapping of HN-Fs to SN-Fs. This mapping can be direct, 3-SN, 5-SN, or 6-SN mode.
2. Ensure that the memory map meets the following requirements:
 - Non-hashed memory regions must not overlap.
 - Hashed memory regions must not overlap.
 - The memory regions must be size-aligned (if the region compare parameters are not enabled).
3. Program the following attributes and registers for each HN-F SAM:

- a. Program the appropriate properties for each SN-F ID, according to the features that are supported in the por_hnf_sam_sn_properties register.
These properties provide the interface width as either 128-bit or 256-bit, CMO support, and PCMO support.
 - b. Program the HN-F to SN-F mapping, which depends on the mapping schemes that are used:
 - If the HN-F is directly mapped to an SN-F, program the SN0 target ID and corresponding attributes.
 - If the HN-F is in 3-SN or 6-SN mode, program the following:
 - All SN-F target IDs and the attributes for each SN-F.
 - The mode of operation as 3-SN or 6-SN.
 - The top address bits.
 - If the HN-F is in 3-SN, 5-SN, or 6-SN mode, program the following:
 - All SN-F target IDs and the attributes for each SN-F.
 - The mode of operation as 3-SN, 5-SN, or 6-SN.
 - The top address bits.
 - If the HN-F uses range-based SN-F partitioning for a particular memory region, program the memory region registers, including the target ID that is associated with each region.
4. Complete the following programming for the RN-F RN SAM:
 - a. Program the following attributes and registers for each SCG:
 - Memory region registers.
 - HN-F count registers.
 - HN-F target ID registers.
 - If PrefetchTgt operations are enabled:
 - SN-F target ID registers for SCG.
 - SN-F target ID selection mode for SCG.
 - If 3-SN or 6-SN mode is enabled, program the top address bits.
 - If 3-SN, 5-SN, or 6-SN mode is enabled, program the top address bits.
 - b. Program the non-hashed memory region registers.
 - c. Program the non-hashed target ID registers.
 - d. Program the rnsam_status register to disable the default target ID mode.
 5. Complete the following programming for each RN-I RN SAM and RN-D RN SAM:
 - a. Program the following registers for each SCG:
 - Memory region registers.
 - HN-F count registers.
 - HN-F target ID registers.
 - b. Program the non-hashed memory region registers.
 - c. Program the non-hashed target ID registers.
 - d. Program the rnsam_status register to disable the default target ID mode.

Example memory map programming

This section describes an example memory map and how to program it in the RN SAM and HN-F SAM.

The following figure shows an example memory map with 1024GB addressable size. It is based on the Arm 40-bit proposed address map. It has three separate DRAM regions (in the address ranges from 2-4GB, 34-64GB and 544-1024GB) and four I/O regions, which must be mapped to specific targets. It is assumed that the I/O region 256-512GB is unused and no requests are sent to this address.

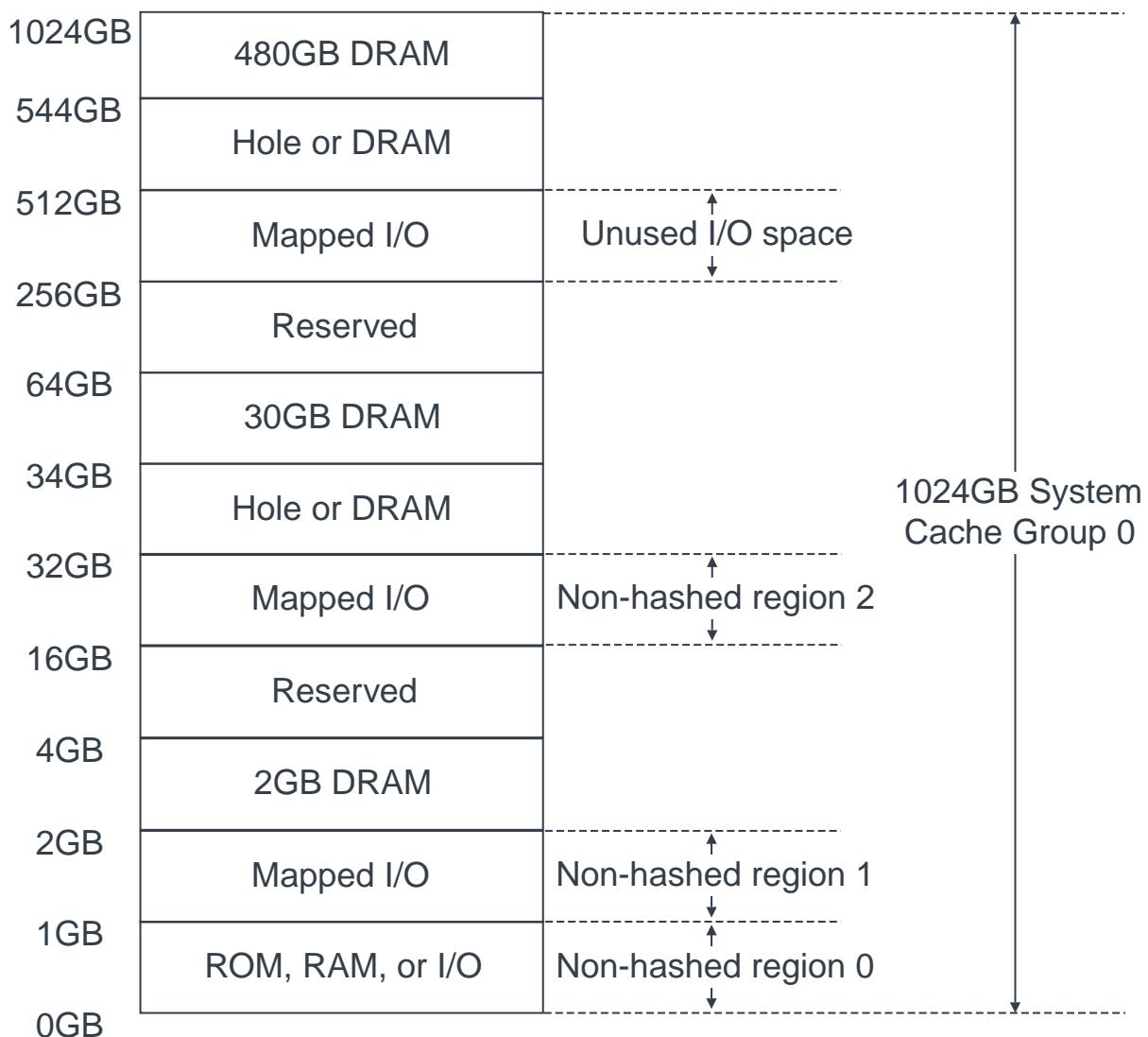


Figure 3-1432 CMN-700 example memory map

It is assumed that there are eight HN-Fs in the system and all HN-Fs are being used for one SCG (group 0). To program the RN SAM, follow these steps:

1. Map the full 1024GB memory map to the system cache group. Arm recommends this mapping because DRAM regions are non-contiguous and the entire DRAM space is assigned to one SCG.
2. Carve out each of the non-hashed regions from the full 1024GB memory map as shown in the preceding figure. Assign each non-hashed region to individual non-hashed targets.
3. When the RN SAM programming is done, turn ON the region-based target ID selection by disabling the default mode of the RN SAM.

The following table shows the RN SAM registers and the corresponding programmed values.

Table 3-1452 RN SAM registers and programmed values

Register name	Field name	Value	Description
sys_cache_grp_region0	region0_base_address	0x0_0000	Base address [51:26]
	region0_size	7'b00001110	1024GB size
	region0_target_type	2'b00	HN-F target type
	region0_valid	1'b1	Region 0 is valid.
sys_cache_grp_hn_nodeid_reg0	nodeid_0	<hnf0_node_id>	Physical node IDs of the HN-Fs in the system from HN-F 0 to HN-F 7.
	nodeid_1	<hnf1_node_id>	If RN-Fs are generating PrefetchTarget operations, then you must program the SN node IDs corresponding to each HN-F in sys_cache_grp_sn_nodeid_regX registers.
	nodeid_2	<hnf2_node_id>	
	nodeid_3	<hnf3_node_id>	
sys_cache_grp_hn_nodeid_reg1	nodeid_4	<hnf4_node_id>	
	nodeid_5	<hnf5_node_id>	
	nodeid_6	<hnf6_node_id>	
	nodeid_7	<hnf7_node_id>	
sys_cache_group_hn_count	scg0_num_hnf	0x08	Total of eight HN-Fs in this system cache group.
non_hash_mem_region_reg0	region0_base_address	0x0_0000	1GB from [51:26] 0x0_0000_0000
	region0_size	7'b00000100	1GB size
	region0_target_type	2'b01	HN-I target type
	region0_valid	1'b1	Region 0 is valid.
non_hash_mem_region_reg1	region1_base_address	0x0_0010	1GB region from 0x0_4000_0000
	region1_size	7'b00000001	1GB size
	region1_target_type	2'b01	HN-I target type
	region1_valid	1'b1	Region 1 is valid.
non_hash_mem_region_reg2	region2_base_address	0x0_0100	16GB region from 0x4_0000_0000
	region2_size	7'b00001000	16GB size
	region2_target_type	2'b01	HN-I target type
	region2_valid	1'b1	Region 2 is valid.
non_hash_tgt_nodeid0	nodeid_0	<hn10_node_id>	Node ID of HN-I 0 corresponding to non-hashed region 0
	nodeid_1	<hn11_node_id>	Node ID of HN-I 1 corresponding to non-hashed region 1
	nodeid_2	<hn12_node_id>	Node ID of HN-I 2 corresponding to non-hashed region 2
rnsam_status	ninstall_req	1'b1	Uninstall any operations that depend on SAM programming.
	default_target	1'b0	Disable default mode and use the programmed ranges for new incoming addresses.

Similarly to RN SAM, HN-F SAM must also be programmed so that it can select the correct SN-F target ID. All HN-F SAMs within SCG 0 must have the same programming, as the following table shows, including the attributes of each SN-F.

Table 3-1453 HN-F programming information

Register name	Field name	Value	Description
por_hnf_sam_control	hn_cfg_sn0_nodeid	<sn0_node_id>	Node ID of SN-F 0
	hn_cfg_sn1_nodeid	<sn1_node_id>	Node ID of SN-F 1
	hn_cfg_sn2_nodeid	<sn2_node_id>	Node ID of SN-F 2
	hn_cfg_three_sn_en	1'b1	Enable 3-SN mode.
	hn_cfg_sam_top_address_bit0	39	Bit 39 of address
	hn_cfg_sam_top_address_bit1	36	Bit 36 of address
	hn_cfg_sam_inv_top_address_bit	1'b1	Invert top address bit.

3.4.4 Program the dual DAT/RSP channel selection scheme

Boot-programmable registers control the dual DAT/RSP channel selection scheme by forming a TgtID LUT. Programming these registers at boot overrides the default channel selection scheme.

When the POR_2XDATRSP_EN_PARAM is enabled, a default channel selection scheme is applied. For more information about the scheme, see [Dual DAT/RSP channel selection on page 2-64](#). This scheme is active until you configure the dual DAT/RSP registers.

A constraint applies to the channel assignment for the RN-F or RN-I and HN-D that is used to program the dual DAT/RSP registers. You must program these nodes to use the same channel as the default channel selection scheme, according to the MXP XID. For example, if the HN-D node is connected to $XP_{(1,0)}$, it has an odd XID (XID = 1), so by default targets channel 1. When programming is complete, the HN-D node ID must have CHN_SEL set to channel 1 in the LUT.

If your configuration uses CALs, then all the devices that are attached to the CAL must use the same DAT/RSP channel. Therefore, you must program the TgtID of both devices into the registers.

Procedure

- Identify and classify the targets that must map to channel 0 and channel 1, for both DAT and RSP channels.
- Program the TGTID and CHN_SEL fields in por_mxp_multi_dat_rsp_chn_sel_* registers with the TgtIDs to map to the required channel, 0 or 1.
- Set the VALID bit in each por_mxp_multi_dat_rsp_chn_sel_* register to indicate that you have configured valid TgtIDs in each register.
- Set the multi_dat_rsp_chn_sel_programmed bit in the por_mxp_multi_dat_rsp_chn_ctrl register to indicate that channel selection configuration is complete.
- Repeat steps 1-4 for each MXP in the mesh and ensure that the same values are programmed in the por_mxp_multi_dat_rsp_chn_sel_* registers for all MXPs.

3.4.5 Program non-XY routing registers

To configure the behavior of the CMN-700 non-XY routing feature at boot, you must program the por_mxp_xy_override_sel_* registers for each MXP.

When you enable this feature by setting the configuration parameter, the default XY route is applied. The default scheme is active until it is reconfigured by programming the por_mxp_xy_override_sel_* registers.

The following constraints apply to this process:

- A maximum of 16 source-target pairs are supported.
- A maximum of eight source-target pairs are supported.
- You must configure all MXPs in your mesh with the same set of <SRCID> and <TGTID> values. These values correspond to the set of source-target pairs that is overridden, so must be identical in all MXPs.

- You must only set the XY override and YX turn bits in the registers for the overridden MXP or MXPs.
- You must only set the XY override bit in the registers for the overridden MXP or MXPs.
- All devices that are attached to a CAL have the same routing scheme applied. Therefore all devices that are attached to a CAL are either part of the non-XY scheme or not. Selection of partial devices behind a CAL for non-XY routing is not allowed.

Prerequisites

You must ensure that your mesh configuration is free of deadlocks when using the non-XY routing feature. For more information on how to avoid deadlocks when setting up this feature, see [Rules for avoiding deadlocks in non-XY routing on page 2-150](#).

Procedure

1. Identify the node source-target pairs and the MXP or MXPs to be overridden.
2. Program the <SRCID> and <TGTID> bits in the relevant por_mxp_xy_override_sel_* registers with the IDs of the chosen source-target pairs.
3. Set the VALID bit in each por_mxp_xy_override_sel_* register to indicate that valid source-target pairs are configured in each register.
4. Repeat steps 1-3 for each MXP in the mesh, ensuring that you program the same values for all MXPs.
5. Set the XY_OVERRIDE_ENABLE bit or bits in any MXPs where you want to override the XY route for a given source-target pair.
6. Set the YX_TURN_ENABLE bit or bits in any MXPs where the YX turn is allowed for a given source-target pair.

3.4.6 RN-I and HN-I PCIe programming sequence

To ensure proper PCIe functionality, software must complete the following programming before any non-configuration access to the RN-I or HN-I.

When setting up PCIe RN-Is and HN-Is, the entire PCIe configuration space of an RC must be mapped to a single HN-I address region. HN-I has a SAM that can be configured for up to three address regions. All address regions that are not configured into these three regions are considered to be the default address region. For more information about configuring the HN-I SAM, including example configurations, see [2.4.9 HN-I SAM on page 2-138](#).

If you map an HN-I SAM address region to a PCIe slave, you must map all address regions of that HN-I SAM to PCIe slaves.

Throughout the following procedure, address region X or address region Y can refer to any of the four address regions (0, 1, 2, or 3).

Note

- Peer-to-peer writes to HN-P always assume that traffic is directed to the PCIe endpoint memory space. Therefore, HN-P does not give early write completions for any write request.
- Peer-to-peer reads to HN-P also assume traffic is directed to PCIe endpoint memory space.

Procedure

1. If there is a PCIe-RC attached to the RN-I then set the pcie_mstr_present field of the por_rni_cfg_ctl register. This programming indicates that one or more PCIe masters are present upstream.
2. Program the por_hni_sam_addrregion{0,1,2,3}_cfg registers so that the PCIe configuration space falls under one of the four HN-I address regions. This programmed address region is referred to as address region X.
3. Set only one of the following bits in por_hni_sam_addrregionX_cfg for address region X:

ser_devne_wr

Set this bit if the PCIe configuration space is marked as the Arm Device-nGnRnE memory type. If this bit is set, HN-I serializes all Device-nGnRnE writes to address region X. The HN-I does not send any other write requests with the same AWID as an outstanding Device-nGnRnE write.

ser_all_wr

Set this bit if the PCIe configuration space cannot be marked as Arm memory type of Device-nGnRnE. If this bit is set, HN-I serializes all writes targeting Address Region X.

4. Clear the pos_early_wr_comp_en bit of the por_hni_sam_addrregionY_cfg register for address region Y.

Address region Y is the region in which PCIe EP memory space (posted traffic) is programmed. If the pos_early_wr_comp_en bit is cleared, HN-I does not give early write completions for any write requests targeting address region Y.

For address region Y, in which EP memory space is programmed, the physical_mem_en field of the por_hni_sam_addrregionY_cfg register can be set to enable Normal memory behavior of posted traffic.

3.4.7 MTSX programming

To ensure that the MTSX generates the correct tag addresses, the MTSX registers must be programmed correctly.

Any misprogramming of MTSX registers might cause UNDEFINED behavior.

Programming MTU tag address generation registers

There are specific programming requirements that you must follow so the MTSX can determine the tag address according to the PA of the request.

You must program all address generation registers consistently with the associated memory controller in the system.

The following registers must be programmed for the MTSX to determine the tag address according to the PA of the request:

- mtu_tag_addr_ctl
- mtu_tag_addr_base
- mtu_tag_addr_shutter{0-2}

mtu_tag_addr_ctl

The mtu_tag_addr_ctl register indicates which memory map mode is being used. A slave that is downstream of MTSX might use a contiguous memory map. However, the regions that the slave serves might be non-contiguous in the CMN-700 SAM. Therefore, the MTSX can use various translation modes to map system addresses to addresses in the memory map of the downstream slave. The memory map mode determines the mapping structure that the MTSX uses to translate these addresses. MTSX supports five memory map modes, which are defined in [por_mtu_tag_addr_ctl](#) on page 3-408.

The following table shows the five supported memory map modes.

Table 3-1454 MTSX memory map modes

Memory map mode	por_mtu_tag_addr_ctl.memory_map_mode encoding	Description
Pass-through	0b000	Used when the system address maps directly to the downstream slave without any translation.
PDD translation	0b001	Used with the PDD memory map. For more information, see Figure 3-1433 PDD memory map on page 3-1247 and the <i>Principles of Arm® Memory Maps White Paper</i> .
Alternate translation	0b010	Used with a memory map that is a mixture of a flat (non-translated) memory map and a PDD-translated memory map. It has a DRAM region between 2GB and 4GB for compatibility with legacy 32-bit peripherals, but is flat for the rest of the map. For more information, see Figure 3-1434 Alternate translation mode memory map on page 3-1248 .
Alternate translation with multiple sockets	0b011	Used with a memory map that is similar to one for which alternate translation mode is used. However, if the MTSX receives an address above 4TB, then it simply drops the most significant bits over 4TB. In this mode, MTSX cannot differentiate between an access to 0x700_0000_0000 and an access to 0x300_0000_0000. Therefore, the system must prevent such accesses. For more information, see Figure 3-1435 Alternate translation with multiple sockets mode memory map on page 3-1249 .
Map type 0 mode	0b100	<p>For more information, see Figure 3-1436 Map type 0 mode memory map on page 3-1250</p> <hr/> <p style="text-align: center;">Note</p> <p>You can only use the map_type_0 encoding with prior written permission from Arm.</p> <hr/>

The following figure shows the PDD memory map, which the MTSX can convert into a contiguous address map using the PDD translation mode.

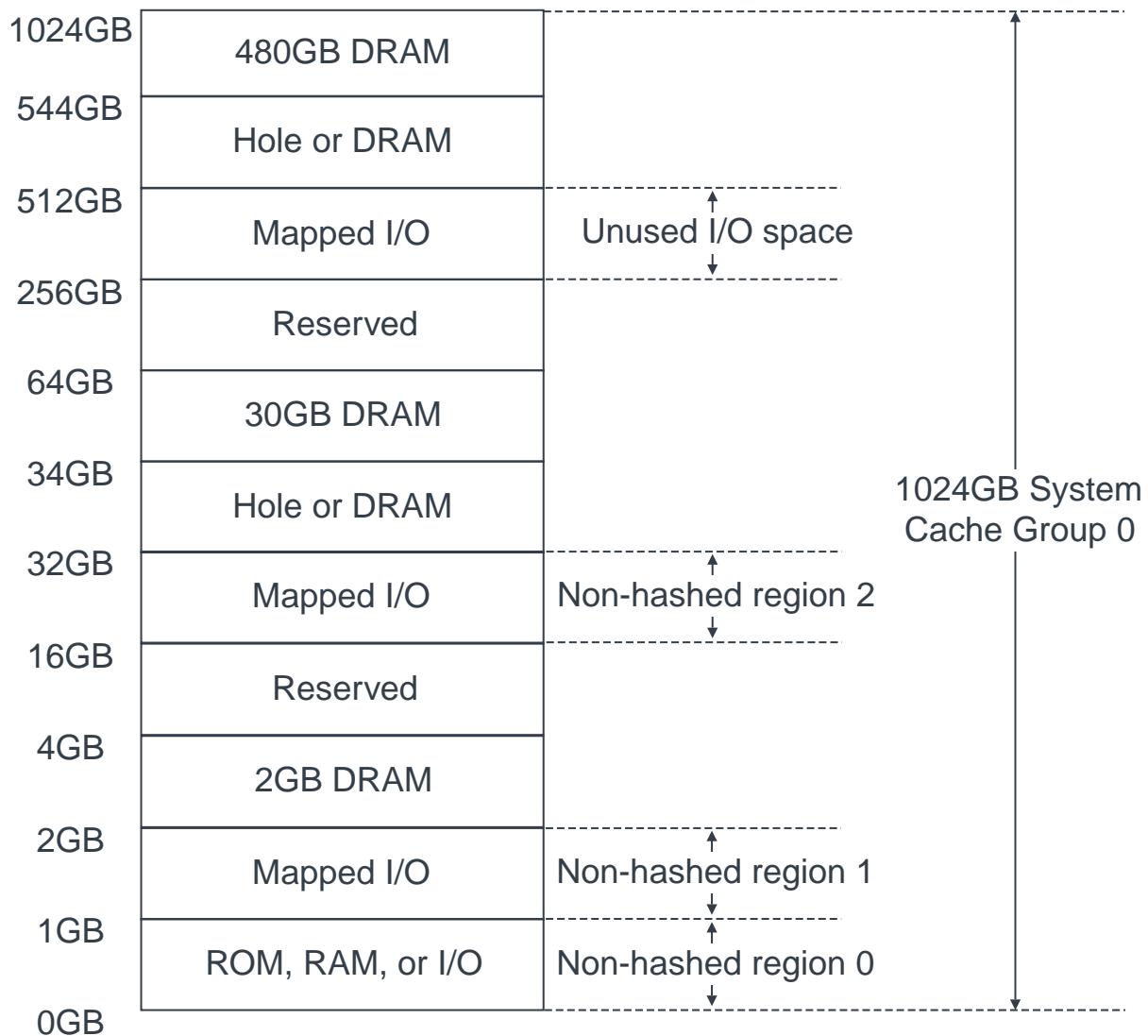


Figure 3-1433 PDD memory map

The following figure shows the memory map that is used with the alternate translation memory map mode.

0xC000_0000_0000

SDRAM
191TB

0x0100_0000_0000

SDRAM
510GB

0x0080_8000_0000

Reserved
510GB

0x0001_0000_0000

SDRAM 2GB

0x0000_8000_0000

Reserved
2GB

0x0000_0000_0000

Figure 3-1434 Alternate translation mode memory map

The following figure shows the memory map that is used with the alternate translation with multiple sockets memory map mode.

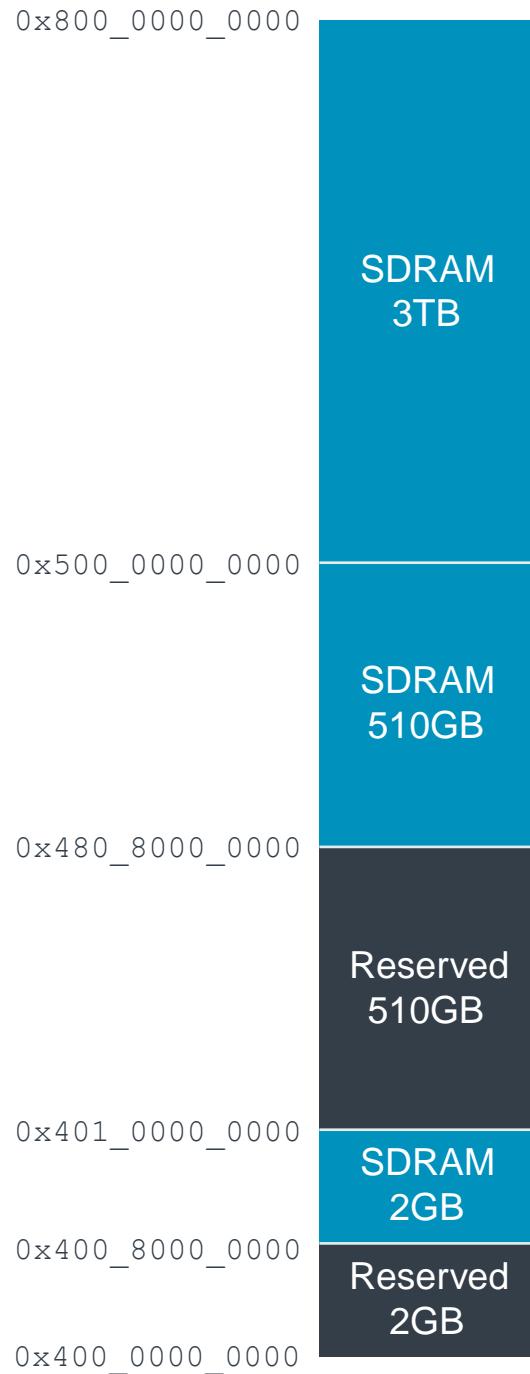


Figure 3-1435 Alternate translation with multiple sockets mode memory map

The following figure shows the memory map that is used with the map type 0 memory map mode.



Figure 3-1436 Map type 0 mode memory map

mtu_tag_addr_base

The mtu_tag_addr_base register indicates the starting PA of the tag space in local DRAM, as shown in the following figure.

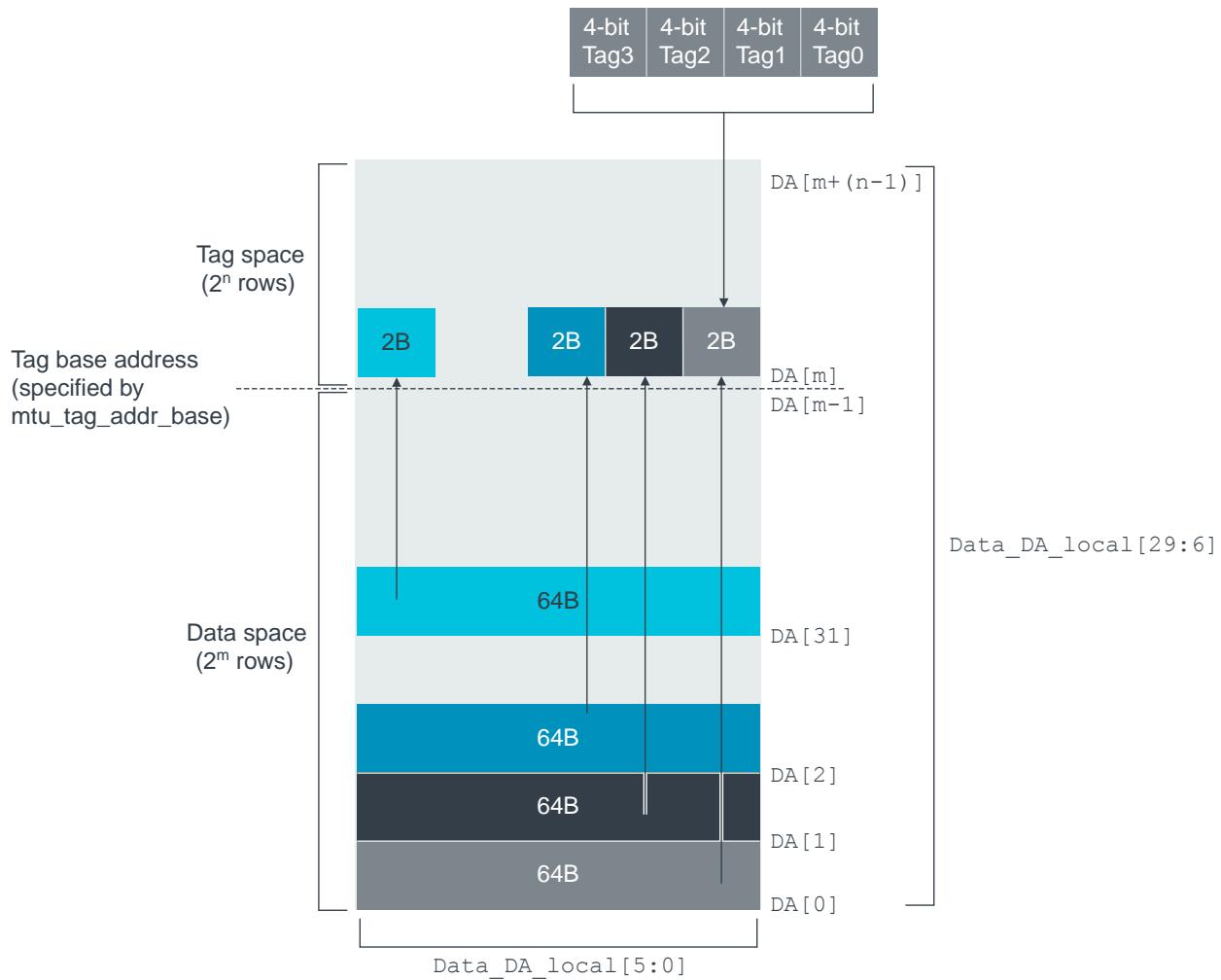


Figure 3-1437 Relationship between mtu_tag_addr_base and DRAM tag space base address

The following constraints apply to the tag regions in DRAM:

- The tag region cannot be interleaved with the data region. The tag region must also be above the data region within DRAM.
- The tag region in the physical address space cannot straddle multiple regions of a memory map.

For example, for the PDD memory map, it is not allowed to have part of the tag region between 2GB-4GB and another part between 34GB-64GB.

mtu_tag_addr_shutter{2:0}

To determine the correct DRAM address, you must program the mtu_tag_addr_shutter{2:0} registers correctly to remove address bits. The address bits that must be removed are based on the number of HNFs and SNs within an SCG, as shown in [Table 2-44 HN-F and SN-F combinations supported within a cache group on page 2-133](#). These registers allow the removal of up to seven PA bits in the range of bit [51] to bit [6]. To program these registers, you must set the shift behavior of each individual address bit as shown in the following figure.

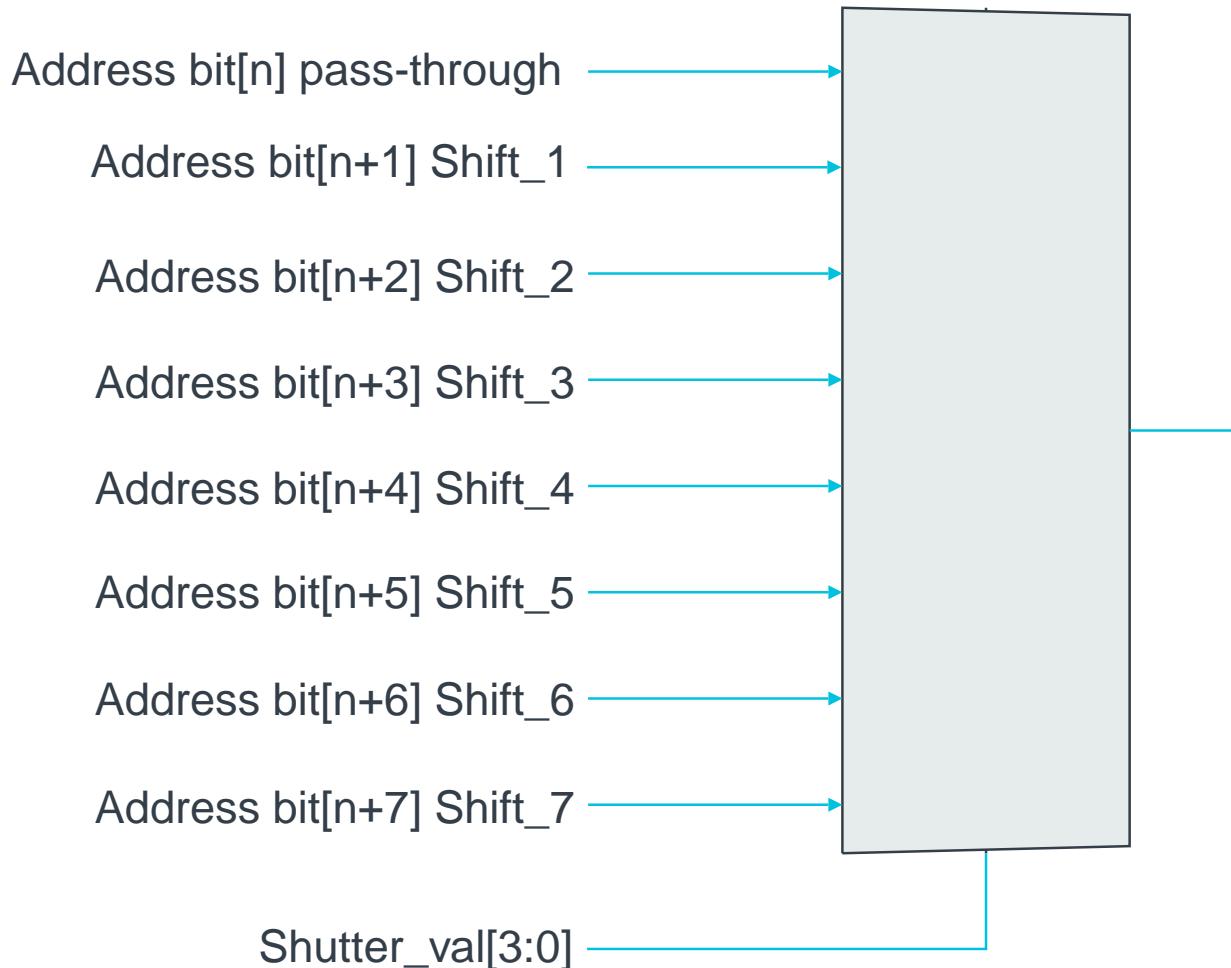


Figure 3-1438 Address shutter register shift behavior

As the preceding figure shows, you can configure the address shutter registers to either:

- Pass through PA bit[n] unchanged.
- Shift another PA bit between bit[n+1] to bit[n+7] into the position of bit[n].

The following figure shows example shutter register programming to remove bit [10] and bit [7] within the PA bit range of bit [12] to bit [6].

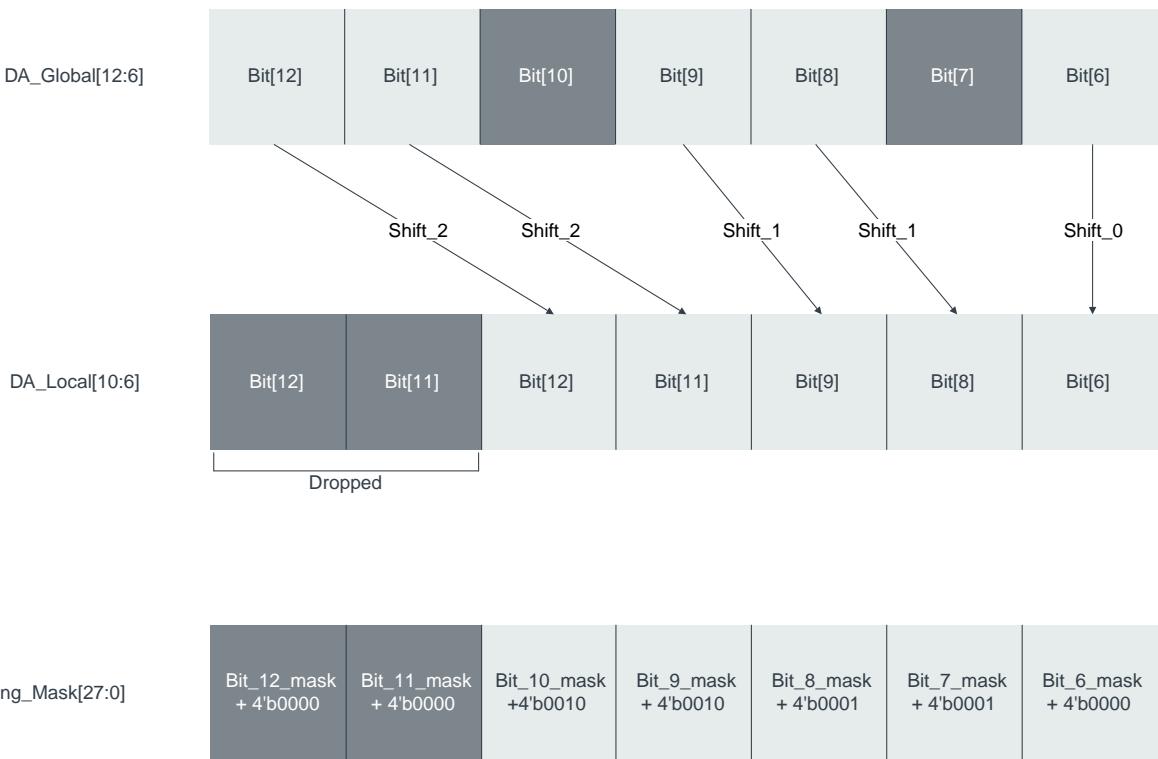


Figure 3-1439 Example address shutter register programming

3.4.8 DT programming

You must follow several programming sequences to set up DTM watchpoints and DTC correctly.

For more information about the CMN-700 DT functionality, see [Chapter 5 Debug Trace and PMU on page 5-1299](#).

Program DTM watchpoint

Use this procedure to program watchpoint N, where N=0..3.

Procedure

1. Program the intended watchpoint matching fields by writing the appropriate values into the por_dtm_wpN_val and por_dtm_wpN_mask registers.
For example, source ID, target ID and opcode are possible matching fields.
2. Program the WP settings in the following register fields to select the device port and flit CHI channel:
 - wp_dev_sel field of the por_dtm_wpN_config register.
 - wp_chn_sel field of the por_dtm_wpN_config register.
3. Program the WP settings in the following register fields to select the device port and flit CHI channel:
 - wp_dev_sel2 and wp_dev_sel fields of the por_dtm_wpN_config register.
 - wp_chn_sel field of the por_dtm_wpN_config register.
4. Program the wp_grp field of the por_dtm_wpN_config register to select primary or secondary watchpoint group.
5. Program the wp_grp field of the por_dtm_wpN_config register to select primary, secondary, or tertiary watchpoint group.
6. Write 1 to the wp_combine field of the port_dtm_wpN_config register if two watchpoints must be combined.

————— Note ————

The wp_combine field is present in WP0 and WP2 only.

7. Program the following register fields if trace packets are to be generated from this watchpoint:
 - a. Program the wp_pkt_type field of the por_dtm_wpN_config register.
 - b. Write 1 to the wp_pkt_gen field of the por_dtm_wpN_config register.
8. Write 1 to the wp_ctrig_en field of the por_dtm_wpN_config register if a cross trigger must be set up from this watchpoint.
9. Program the wp_dbgtrig_en field of the por_dtm_wpN_config register if a debug watchpoint trigger must be set up from this watchpoint.
10. Set the wp_cc_en field of the por_dtm_wpN_config = 1 if a cycle count is needed in the trace packet.
11. Write 1 to the trace_tag_enable field of the por_dtm_control register if a debug watchpoint trace tag must be generated.
12. Write 1 to the dtm_enable field of the por_dtm_control register to enable the WP.

Program DTC

Use this procedure to set up the CMN-700 debug trace control functionality.

The **NIDEN** input signal must be asserted for any trace and PMU operations. Before trace and PMU operations can occur, you must first program and enable watchpoint functions in the DTMs.

The following registers and register bits are present only in the main DTC (DTC0):

- por_dt_secure_access register
- dt_en field of the por_dt_dtc_ctl register
- wait_for_trigger field of the por_dt_dtc_ctl register
- cc_start field of the por_dt_dtc_ctl register
- pmu_en field of the por_dt_pmcr register
- por_dt_pmsrr register
- ss_cfg_active field of the por_dt_pmssr register
- ss_pin_active field of the por_dt_pmssr register

Procedure

1. Write 1 to the dbgtrigger_en field of the por_dt_dtc_ctl register if DBGWATCHTRIG must be generated for DTM debug watchpoint trigger.
2. Write 1 to the atbtrigger_en field of the por_dt_dtc_ctl register if ATB trigger must be generated for DTM debug watchpoint trigger.
3. Write 1 to the cc_enable field of the por_dt_trace_control register to enable cycle count.
4. Write 0 to the dt_wait_for_trigger field of the por_dt_dtc_ctl register if no cross trigger is required.
5. Write 1 to the dt_en field of the por_dt_dtc_ctl register.

3.4.9 PMU system programming

You must follow specific programming sequences to set up the PMU, PMU snapshot, and PMU interrupt functionality correctly.

Set up PMU counters

Use this procedure to set up the PMU counters correctly.

Procedure

1. Ensure that the **NIDEN** input is asserted for any trace and PMU operations.
2. Program (dev)_pmu_event_sel register in the devices or XP.
3. Program the pmevcnt{0..3}_input_sel fields of the por_dtm_pmu_config register to select PMU event counter inputs.

The input can be from one of the following:

- A watchpoint.
- Selected events from the devices or XP, depending on step 2.

4. Program the pmevcnt{0..5}_input_sel fields of the por_dtm_pmu_config register to select PMU event counter inputs.

The input can be from one of the following:

- A watchpoint.
- Selected events from the devices or XP, depending on step 2.

5. Program the following por_dtm_pmu_config register fields to select the paired top global PMU counters:

- pmevcnt_paired
- pmevcnt{0..3}_global_num

6. Program the pmevcnt{01, 23}_combined fields of the por_dtm_pmu_config for any combined local PMU counters.

7. Write 1 to the pmu_en field of the dtm_pmu_config register.

————— Note ————

To activate CXLA PMU function, program the en_cxla_pmucmd_prop field of the associated por_cxg_ra_cfg_ctl register to 1.

8. Program the cntcfg field of the por_dt_pmcr register to pair the 32-bit global counters to make a 64-bit counter.

9. Write 1 to the ovfl_intr_en field of the por_dt_pmcr register to enable interrupts on **INTREQPMU** on any global counter overflow.

10. Write 1 to the pmu_en field of the por_dt_pmcr register to start PMU operation.

Program PMU snapshot

Use this procedure to set up the PMU snapshot functionality.

For a system with multiple DTCs, the sub-DTC maintains snapshot status for the DTM within its own domain.

Prerequisites

The NIDEN input must be asserted for any trace and PMU operation.

Procedure

1. Program PMU counters as described in [Set up PMU counters on page 3-1254](#).

2. Write 1 to the ss_req field of the por_dt_pmsrr register.

This action causes the DTC to send a PMU snapshot instruction. On receiving this instruction, the DTM sends PMU snapshot packets to the DTC.

The DTC updates the ss_status field of the por_dt_pmssr register after receiving PMU snapshot packets. Software can poll this register field to check if the snapshot process is complete.

Program PMU counter overflow interrupt

Use this procedure to set up the PMU counter overflow interrupt.

Prerequisites

The NIDEN input must be asserted for any trace and PMU operation.

Procedure

1. Program PMU counters as described in [Set up PMU counters on page 3-1254](#).

2. Write 1 to the ovfl_intr_en field of the por_dt_pmcr register.
Results: Overflow of any PMU counter causes **INTREQPMU** to assert.
3. Write 1 to the ovfl_intr_en field in the all other por_dt_pmcr registers if your system has multiple DTCs.
4. Poll the pmovsr[7:0] field of the por_dt_pmovsr register when **INTREQPMU** is asserted to see which global counter causes the interrupt.
For multiple DTCs, all por_dt_pmovsr registers must be polled.
5. Write 1 to the corresponding bit in the pmovsr_clr[7:0] field of the por_dt_pmovsr_clr register to clear **INTREQPMU**.

3.5 CML programming

The system must be programmed to enable correct operation with CML.

————— Note —————

Programming steps specified in "CML programming" section are applicable to both CXG and CCG units. Any differences will be specified in the respective section. CCRA and CCHA registers should be same as CXRA and CXHA and corresponding CCG registers must be used. CCLA registers are different, therefore registers and corresponding programming steps will be specified separately where needed.

This section contains the following subsections:

- [3.5.1 CML-related programmable registers on page 3-1257](#).
- [3.5.2 Bring up a CML system on page 3-1258](#).
- [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).
- [3.5.4 Program CMN-700 CML system at runtime on page 3-1268](#).
- [3.5.5 Establish protocol link up between CXG and remote CCIX link on page 3-1269](#).
- [3.5.6 Link down CCIX protocol link between CXG and remote CCIX link on page 3-1269](#).
- [3.5.7 CCIX entry and exit protocol links from coherency domains and DVM domains on page 3-1270](#).

3.5.1 CML-related programmable registers

This section contains a list of CML programmable registers.

CXRA

- RA SAM address region registers (por_cxg_ra_sam_addr_region_reg<X>)
- LDID to RAID LUT registers:
 - por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg<X>
 - por_cxg_ra_rnf_ldid_to_nodeid_reg<X>
 - por_cxg_ra_rnf_ldid_to_exp_raid_reg<X>
 - por_cxg_ra_rni_ldid_to_exp_raid_reg<X>
 - por_cxg_ra_rnd_ldid_to_exp_raid_reg<X>
- RAID or HAID to LinkID LUT registers:
 - por_cxg_ra_agentid_to_linkid_reg<X>
 - por_cxg_ra_agentid_to_linkid_val
- CCIX Protocol Link Control and Status registers:
 - por_cxg_ra_cxrptcl_link<X>_ctl
 - por_cxg_ra_cxrptcl_link<X>_status
- Auxiliary Control register (por_cxg_ra_aux_ctl)
- Configuration Control register (por_cxg_ra_cfg_ctl)
- PortID assignment (por_cxg_ra_node_info)

CXHA

- HAID register (por_cxg_ha_id)
- RAID to LDID LUT registers
 - por_cxg_ha_rnf_exp_raid_to_ldid_reg<X>
- RAID or HAID to LinkID LUT registers:
 - por_cxg_ha_agentid_to_linkid_reg<X>
 - por_cxg_ha_agentid_to_linkid_val
- CCIX Protocol Link Control and Status registers:
 - por_cxg_ha_cxrptcl_link<X>_ctl
 - por_cxg_ha_cxrptcl_link<X>_status
- Auxiliary Control register (por_cxg_ha_aux_ctl)
- RN SAM

CXL A

- CCIX capabilities (por_cxla_ccix_prop_capabilities). This register is RO.
- CCIX Configured Properties (por_cxla_ccix_prop_configured)
- CXS Interface Properties Registers. These registers are RO:
 - por_cxla_tx_cxs_attr_capabilities
 - por_cxla_rx_cxs_attr_capabilities
- RAID or HAID to LinkID LUT registers:
 - por_cxla_agentid_to_linkid_reg<X>
 - por_cxla_agentid_to_linkid_val
- LinkID to PCIe Bus Number LUT register (por_cxla_linkid_to_pcie_bus_num)
- Auxiliary Control register (por_cxla_aux_ctl)
- Port enable and AgentID to PortID functionality:
 - por_cxla_portfwd_ctl
 - por_cxla_agentid_to_portid_reg<X>
- TLP header field value configuration register (por_cxla_tlp_hdr_fields)

HN-F

- LDID to CHI NodeID registers (por_hnf_rm_phys_id<X>)
- CCIX Port Aggregation Mask register (por_hnf_cml_port_aggr_grp0_add_mask)
- CCIX Port Aggregation Control register (por_hnf_cml_port_aggr_grp0_reg)
- LDID to CHI NodeID registers (por_hnf_rm_cluster<X>.physid_reg<Y>)
- CCIX Port Aggregation Mask register (por_hnf_cml_port_aggr_grp<X>.add_mask)
- CCIX Port Aggregation Control register (por_hnf_cml_port_aggr_grp<X>.reg)

RN-F/RN-I/RN-D

- RN SAM
- CCIX Port Aggregation Mode Enable and Control registers:
 - cml_port_aggr_grp0_reg
 - cml_port_aggr_mode_ctrl_reg
- CCIX Port Aggregation Mask register (cml_port_aggr_grp0_add_mask)
- RN SAM
- CCIX Port Aggregation Mode Enable and Control registers:
 - cml_port_aggr_grp<X>.reg
 - cml_port_aggr_mode_ctrl_reg
- CCIX Port Aggregation Mask register (cml_port_aggr_grp<X>.add_mask)

3.5.2 Bring up a CML system

Use the following sequence to bring up a CML system.

Procedure

1. Discover and bring up the local CMN-700 system.
For more information, see [Discover and bring up a local CMN-700 system](#) on page 3-1259.
2. Discover CCIX devices and CCIX systems.
For more information, see [Discover CCIX devices in CCIX system](#) on page 3-1259.
3. Enumerate and configure CCIX devices.
For more information, see [Enumerate and configure CCIX devices](#) on page 3-1260.

Next Steps

For information about the programming requirements that are necessary for communication between CCIX components, see [3.5.3 Program CML system to enable CCIX communication](#) on page 3-1260.

Discover and bring up a local CMN-700 system

Use this process to bring up a local CMN-700 system during a full bring up of a CML system.

This procedure is the first step in the sequence to bring up a CMN-700 CML system. For the full sequence, see [3.5.2 Bring up a CML system on page 3-1258](#).

Procedure

1. Complete the CMN-700 discovery mechanism to discover node types, their corresponding locations (node IDs), and their logical IDs.
[2.4.1 Node ID mapping on page 2-95](#) and [2.5 Discovery on page 2-163](#) define the discovery mechanism. Node types that are relevant to CML-specific programming are RN-Fs, RN-Is, RN-Ds, HN-Fs, and CML gateway blocks (CXRA, CXHA, and CXLA).
2. Bring up the local system to allow normal local operations. To bring up all local non-CML components (HN-F, HN-D, HN-I, RN-I, SN-F, and XP):
 - a. Complete CMN-700 boot time programming.
For more information, see [3.4.1 Boot-time programming sequence on page 3-1238](#).
 - b. Program RN SAM with the local address map.
For more information, see [Program the SAM on page 3-1239](#).
3. Bring up the local system to allow normal local operations. To bring up all local non-CML components (HN-F, HN-I, HN-D, HN-P, RN-I, SN-F, and XP):
 - a. Complete CMN-700 boot time programming.
For more information, see [3.4.1 Boot-time programming sequence on page 3-1238](#).
 - b. Program RN SAM with the local address map.
For more information, see [Program the SAM on page 3-1239](#).

Next Steps

For information about how to discover the CCIX devices in your CML configuration, see [Discover CCIX devices in CCIX system on page 3-1259](#).

Discover CCIX devices in CCIX system

CCIX system discovery can involve going through the PCIe link activation and device enumeration mechanism. Use this process to discover CCIX devices in your CCIX system.

This procedure is the second step in the sequence to bring up a CMN-700 CML system. For the full sequence, see [3.5.2 Bring up a CML system on page 3-1258](#).

Prerequisites

You must first discover and bring up your local CMN-700 system. For more information, see [Discover and bring up a local CMN-700 system on page 3-1259](#).

Procedure

- Follow the standard PCIe device enumeration steps to detect CCIX capable devices.
For more information, see *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*.

Note

PCIe and CCIX device enumeration steps can be skipped for SMP connection.

Next Steps

If one or more CCIX-capable devices are detected during PCIe device enumeration, then complete the following steps:

- See [Enumerate and configure CCIX devices on page 3-1260](#), which shows how to enumerate CCIX devices.
- Program the PCIe-RC to enable multiple *Virtual Channels* (VCs).

Enumerate and configure CCIX devices

Use this procedure to enumerate CCIX devices in your CML system and configure their CCIX properties during bring up.

This procedure is the last step in the sequence to bring up a CMN-700 CML system. For the full sequence, see [3.5.2 Bring up a CML system on page 3-1258](#).

Prerequisites

Before enumerating CCIX devices, you must first complete the following steps:

- Discover and bring up your local CMN-700 system. For more information, see [Discover and bring up a local CMN-700 system on page 3-1259](#).
- Discover CCIX devices in your CCIX system. For more information, see [Discover CCIX devices in CCIX system on page 3-1259](#).

Procedure

1. Discover all CCIX agents (RA and HA) at each CCIX device.

These agents must be uniquely identifiable. If any CCIX device contains CMN-700, follow the CMN-700 discovery mechanism to discover node types, their corresponding locations (node IDs), and their logical IDs. The discovery mechanism is described in [2.4.1 Node ID mapping on page 2-95](#) and [2.5 Discovery on page 2-163](#).

2. Discover the address map requirements of each CCIX device.
3. Read the CCIX capabilities of each CCIX device.
 - a. Read the por_cxla_ccix_prop_capabilities register, which is present in each CXLA, to determine the CCIX capabilities of CMN-700.
4. Determine the common properties and capabilities that all CCIX devices support and configure them in each CCIX device.
5. Program the properties that are determined in the preceding step in the por_cxla_ccix_prop_configured register, which is present in each CMN-700 CXLA.

Next Steps

To enable CCIX communication, follow the programming procedures that are described in [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).

3.5.3 Program CML system to enable CCIX communication

Use this procedure to enable CCIX communication between different CCIX entities. The steps can be completed in any order.

The terms *link*, *CCIX link*, and *CCIX protocol link* that are used in subsequent sections refer to CCIX logical link.

The CCIX logical link is defined in the *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*.

Prerequisites

Before enabling CCIX communication, you must first complete the bring up process. For more information, see [3.5.2 Bring up a CML system on page 3-1258](#).

Procedure

- Program the auxiliary control and configuration control registers.

For more information about specific optional functionality, see [Options when programming CXG auxiliary control and configuration control registers](#) on page 3-1261.

- Program IDs for local CXRAs and CXHAs.
For more information, see [Assign IDs for local CXRAs and CXHAs](#) on page 3-1261.
- Program remote CCIX agents:
 - Assign LinkIDs to remote CCIX protocol links. For more information, see [Assign LinkIDs to remote CCIX protocol links](#) on page 3-1262.
 - Assign PCIe bus numbers for LinkIDs. For more information, see [Assign PCIe bus numbers for LinkIDs](#) on page 3-1263.
 - Assign LDIDs to remote caching agents. For more information, see [Assign LDIDs to remote caching agents](#) on page 3-1263.
 - Program RA SAMs. For more information, see [Program RA SAM](#) on page 3-1264.
 - Program RNSAM in CXHAs. For more information, see [Program RN SAM in CXHA](#) on page 3-1264.
 - Program CCIX protocol link control registers. For more information, see [Program CCIX protocol link control registers](#) on page 3-1264.
 - Program CPA functionality in RN SAM, if using CPAGs. For more information, see [Program CPA functionality in RN SAM](#) on page 3-1265.
 - Program CPA functionality in HN-F SAM, if using CPAGs. For more information, see [Program CPA functionality in HN-F SAM](#) on page 3-1266.
- Enable CCIX port-to-port forwarding, if using this feature.
For more information, see [Enable CCIX port-to-port forwarding](#) on page 3-1267.

Options when programming CXG auxiliary control and configuration control registers

CMN-700 has optional CML functionality that can be enabled when you program the CXG auxiliary control and configuration control registers. There are specific constraints that you must follow when enabling this functionality for your CML system.

CMN-700 supports the following functionality in the CXG auxiliary and configuration control registers:

- SMP mode
- CXSA mode

SMP mode

SMP connection requires CCG block and is enabled by setting the Lnk<X>_smp_mode_en bit in the following registers:

- por_ccg_ra_exptrcl_link<X>_ctl
- por_ccg_ha_exptrcl_link<X>_ctl

The SMP mode programming must be the same in CCRA and CCHA for a specific CCIX protocol link. Also, all CCG pairs that can communicate with each other must be configured in the same way.

CXSA mode

CXSA connection requires CXG block and is enabled by setting the cxsa_mode_en bit in the por_cxg_ra_cfg_ctl register of the CXRA.

When this mode is enabled, the CXRA inside the CXG is used to communicate with a remote CXSA. In this mode, the CXRA receives requests from local HN-Fs.

Assign IDs for local CXRAs and CXHAs

Use this procedure to assign CCIX identifiers for local CXRAs and CXHAs and configure them in the relevant registers.

This task is part of the requirements to enable communication between local and remote CCIX agents. For the full list of requirements, see [3.5.3 Program CML system to enable CCIX communication](#) on page 3-1260.

Procedure

- Assign *Requesting Agent IDs* (RAIDs) for local CXRAs.
 1. Program all the local RAIDs in the following registers for all CXRAs and set the corresponding valid bit in each CXRA:
 - por_cxg_ra_rmf_ldid_to_exp_raid_reg
 - por_cxg_ra_rni_ldid_to_exp_raid_reg
 - por_cxg_ra_rnd_ldid_to_exp_raid_reg
 2. Program the CCIX Source ID (Agent ID) in entry 0 of the por_cxg_ra_rmf_ldid_to_exp_raid_reg register and set the corresponding valid bit if CXSA mode is enabled.
- Program *Home Agent IDs* (HAIDs) for all local CXHAs into the por_cxg_ha_id registers that are present in each CXHA.

This programming is not required if CXSA mode is enabled.

Because all CXHAs can communicate with all local HNs (HN-F, HN-I, HN-D, and HN-P), they can have the same HAID. However, if uniqueness between HAIDs is required for routing purposes, HAIDs do not have to be the same.

For compliance with the CCIX specification, a CXHA and CXRA with the same ID must reside behind the same CCIX protocol link. For more information, see *Cache Coherent Interconnect for Accelerators CCIX Base Specification Revision 1.1 Version 1.0*.

Assign LinkIDs to remote CCIX protocol links

Use this procedure to assign a unique LinkID to each remote CCIX protocol link with which a CXG can communicate.

————— Note —————

CCG supports only one link and all valid agents must be programmed with LinkID 0.

This task is part of the requirements to enable communication between local and remote CCIX agents. For the full list of requirements, see [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).

Each CMN-700 CXG contains a CXRA, CXHA, and CXLA node. Each CXG can communicate with up to three remote CCIX protocol links. These links are marked sequentially as links 0, 1, and 2. Remote links are identified using LinkIDs.

Remote CCIX agents (RAs or HAs) are identified using their RAID or HAID. Each remote RA or HA that a CXG can communicate with must be behind only one link.

It is only necessary for LinkIDs to be unique within a CXG. Each CXG has a respective LinkID space. Each remote link has its own CCIX protocol link control and status registers.

Procedure

1. Determine the LinkID of each remote agent, in other words the targets, of the CXG.
2. Program these LinkIDs in the following registers, which are present in the CXRA, CXHA, and CXLA:
 - por_cxg_ra_agentid_to_linkid_reg<X>
 - por_cxg_ha_agentid_to_linkid_reg<X>
 - por_cxla_agentid_to_linkid_reg<X>

This step sets up the AgentID (RAID or HAID) to LinkID LUT.

If CXSA mode is enabled for a link, you do not need to program the por_cxg_ha_agentid_to_linkid_reg<X> register. The por_cxg_ra_agentid_to_linkid_reg<X> register must be programmed with the CCIX Slave Agent ID for the link.

3. Set the respective valid bits in the following registers:
 - por_cxg_ra_agentid_to_linkid_val
 - por_cxg_ha_agentid_to_linkid_val
 - por_cxla_agentid_to_linkid_val

Program AgentID to PortID sets

Use this procedure to program AgentID to PortID sets.

————— Note —————

Port-to-Port forwarding is supported in CXG block and therefore the following AgentID to PortID programming is valid only for CXG.

Procedure

1. Determine the target PortID for each target AgentID (CCIX TLP TargetID) that the current port can forward the incoming CCIX TLPs to.
2. Program these PortIDs in the por_cxla_agentid_to_portid_reg<X> register.
For example, all agents which are local to this chip are assigned the same PortID as the current CCIX port. The LUT indicates that the transaction target is on the current chip, so all the TLPs targeting these agents are consumed locally. They are not forwarded to another port.

Assign PCIe bus numbers for LinkIDs

Use this procedure to set up the LinkID to PCIe bus number LUT in the CXLA. This programming is only required if the PCIe header is used to route a CCIX. This step is not needed for CCG block.

This task is part of the requirements to enable communication between local and remote CCIX agents. For the full list of requirements, see [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).

Procedure

- Program the PCIe bus number for each remote link in the por_cxla_linkid_to_pcie_bus_num register, which is present in each CXLA.
- Program the por_cxla_tlp_hdr_fields register if using PCIe headers to route a CCIX TLP.

Assign LDIDs to remote caching agents

Use this procedure to assign a unique LDID for each remote caching agent (RN-F) that can send requests to HNs (HN-F, HN-I, HN-D, and HN-P).

This task is part of the requirements to enable communication between local and remote CCIX agents. For the full list of requirements, see [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).

HN-Fs use the LDID of remote caching agents for SF tracking. LDID assignment is not required for non-caching RAs, for which snooping is not required.

This programming is not required if CXSA mode is enabled.

Procedure

1. Program unique LDIDs for each remote caching agent in the por_cxg_ha_rnf_exp_raid_to_ldid_reg<X> register that is present in each CXHA.
The LDID values for remote RN-Fs must be greater than those values that are used by the local RN-F nodes, unless you override local RN-F LDIDs by using the por_cxg_ra_rnf_ldid_to_ovrd_ldid_reg_0-127 register.
2. Set the ldid<X>_rfn bit, which marks the remote agent as a caching agent, and set the respective valid bit.

3. Program the CXHA NodeID at the LDID index of each remote RN-F in por_hnf_rm_cluster<X>_physid_reg<Y> register in the HN-F. Set the appropriate attributes for remote bit, CPA enable, and CPA group information for each LDID.

Program RA SAM

Use this procedure to program the RA SAM, which generates the target ID for CCIX requests.

This task is part of the requirements to enable communication between local and remote CCIX agents. For the full list of requirements, see [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).

Procedure

- Program the following properties for each remote HA into the por_cxg_ra_sam_addr_region_reg<X> register that is present in each CXRA, and set the corresponding valid bit.
 - The base address of the address region and the corresponding size of the address region
 - The HAID that requests for the address range are mapped toIf CXSA mode is enabled, RA SAM must be programmed with the address range and the Agent ID of the remote CCIX Slave Agent.

Program RN SAM in CXHA

Use this procedure to program the RN SAM in each CXHA.

You can program CXHA RN SAM as part of local system bring up.

This programming is not needed if CXSA mode is enabled.

Procedure

- Program the RN SAM present in each CXHA with the address and memory map of the local HNs. For more information on RN SAM programming, see [3.4.3 RN SAM and HN-F SAM programming on page 3-1239](#).

Program CCIX protocol link control registers

Use this procedure to set up CCIX protocol links for a CXG and configure the distribution of credits that the CXG uses.

There is a CCIX protocol link control register for each CCIX protocol link that a given CCIX gateway block (CXRA, CXHA, and CXLA) can communicate with.

Procedure

1. Program the por_cxg_ra_exprtcl_link<X>_ctl and por_cxg_ha_exprtcl_link<X>_ctl registers that are present in each CXRA and CXHA.
If CXSA mode is enabled, the protocol link control registers of the CXHA do not need to be programmed. In this case, the lnk0_num_snpcrds of the CXRA can be set to 4'hF.
2. Set the lnk<X>_link_en bit for each CCIX protocol link that can be used in the future.
If this bit is not set, credits are not set aside for this link.
3. Program the lnk<X>_num_{snpcrds, reqcrds, daterds} fields with the percentage of protocol credits that must be assigned or granted for a given link.
This step is optional. Default credits are equally assigned or granted to each enabled link as determined by the link enable bit (lnk<X>_link_en). For more information about credit distribution and the permitted configurations, see [CCIX protocol link credit distribution on page 3-1265](#).

You must ensure that the total percentage of credits that are allocated to all links does not exceed 100.

CCIX protocol link credit distribution

When setting up CCIX protocol links, you can specify the CCIX protocol link credit distribution. The distribution can be configured when programming the CCIX protocol link control registers.

The link enable bits (`lnk<X>_link_en`) of the `por_cxg_ra_cxrptcl_link<X>_ctl` and `por_cxg_ha_cxrptcl_link<X>_ctl` registers determine how many links are active for a CXG. By default, credits are equally assigned or granted to each enabled link. However, you can program these registers to configure the distribution of credits across multiple links.

The following table shows the number of links and allowed credit distribution percentages for that number of links.

Table 3-1455 Number of links and allowed credit distribution percentage

Number of links	Allowed credit distribution
1	100%
2	50% : 50%
	25% : 75%
3	50% : 25% : 25%
	33% : 33% : 33%

After distributing credits based on the programmed percentage across all the links available, any remaining credits are allocated to link0. For example, link0 is allocated 44 credits while link1 and link2 are allocated 42 credits each for the 128 credit, 33% credit distribution configuration.

For more information about programming the CCIX protocol link control registers, see [Program CCIX protocol link control registers on page 3-1264](#).

Program CPA functionality in RN SAM

There is a specific sequence of programming steps that must be followed to set up the RN SAM to distribute requests to CPAGs.

This task is part of the requirements to enable communication between local and remote CCIX agents. For the full list of requirements, see [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).

To enable CPA, you must program registers in both the RN SAM and HN-F SAM. For the programming sequence for the HN-F SAM registers, see [Program CPA functionality in HN-F SAM on page 3-1266](#).

There are certain rules and restrictions that apply to how CPA can be enabled in a CMN-700 system. For more information, see [SAM support for CCIX Port Aggregation on page 2-122](#).

Note

The CPA group ID control bits for non-hashed region 7 are aligned differently from other regions. For more information, see the index of the `region7_pag_grpid` field in [cml_port_aggr_mode_ctrl_reg on page 3-649](#).

In a two-chip system with CPA enabled, the RN SAM CPA mask of Chip 0 and HN-F SAM CPA mask of Chip 1 must be programmed to match. Similarly, the RN SAM CPA mask of Chip 1 must match the HN-F SAM CPA mask of Chip 0.

Procedure

- Set the `region<n>_pag_en` field of the `cml_port_aggr_mode_ctrl_reg` register to 1 for each non-hashed memory region that belongs to a remote chip and is required to use CPA.

This programming enables CPA mode for the non-hashed memory region and specifies the CPAG that requests must be hashed across.

2. Set the region<n>_pag_en field of the cml_port_aggr_mode_ctrl_reg register to 1 and region<n>_pag_grpid with the relevant group ID for each non-hashed memory region that belongs to a remote chip and is required to use CPA.

This programming enables CPA mode for the non-hashed memory region and specifies the CPAG that requests must be hashed across.

3. Set each bit that must be used to hash requests across CXGs to 1 in the addr_mask field of the cml_port_aggr_grp<n>_addr_mask register. Set each bit that must be masked from the hash function to 0.

If CPA mode is enabled, RN SAM hashes PA bits[47:6] to distribute traffic between CXGs. The PA is compared against the addr_mask field of the cml_port_aggr_grp<n>_addr_mask_register. Any PA bit that corresponds to a bit with a 0 value in the mask register is masked from hashing.

If CPA mode is enabled, RN SAM hashes PA bits[51:6] to distribute traffic between CXGs. The PA is compared against the addr_mask field of the cml_port_aggr_grp<n>_addr_mask_register. Any PA bit that corresponds to a bit with a 0 value in the mask register is masked from hashing.

4. Program the characteristics of the CXGs for each CPAG by completing the following steps:
 - a. Program the number of CXGs in each CPAG by setting the num_cxg_pag0 field in the cml_port_aggr_grp<m> register.
 - b. Program the CHI node ID of each CXG by setting the pag_tgtid<n> fields in cml_port_aggr_grp_reg0 and cml_port_aggr_grp_reg1.
 - a. Program the number of CXGs in each CPAG by setting the num_cxg_pag<n> field in the cml_port_aggr_ctrl_reg register.
 - b. Program the CHI node ID of each CXG by setting the pag0_tgtid field of the same register.

If Two CXGs are being used, then both node IDs must be programmed in this register.

This programming is used by all regions that have CPA enabled.

5. Repeat the preceding steps for all RN SAMs in the chip.

Program CPA functionality in HN-F SAM

There is a specific sequence of programming steps that must be followed to set up the HN-F SAM to distribute snoop traffic to CPAGs.

This task is part of the requirements to enable communication between local and remote CCIX agents. For the full list of requirements, see [3.5.3 Program CML system to enable CCIX communication on page 3-1260](#).

To enable CPA, you must program registers in both the RN SAM and HN-F SAM. For the programming sequence for the RN SAM registers, see [Program CPA functionality in RN SAM on page 3-1265](#).

There are certain rules and restrictions that apply to how CPA can be enabled in a CMN-700 system. For more information, see [SAM support for CCIX Port Aggregation on page 2-122](#).

In a two-chip system with CPA enabled, the RN SAM CPA mask of Chip 0 and HN-F SAM CPA mask of Chip 1 must be programmed to match. Similarly, the RN SAM CPA mask of Chip 1 must match the HN-F SAM CPA mask of Chip 0.

Procedure

1. Set the following fields in the por_hnf_rn_cluster<X>_physid_reg<Y> registers for each valid LDID in the system:

nodeid_lid<X>_ra<Y>

When using CPA, set this value to match the pag_tgtid0 field of the por_hnf_cml_port_aggr_grp_reg<m> field of the corresponding CPAG.

remote_lid<X>_ra<Y>

Set this value to 1 if the RN-F is a remote requestor.

cpa_grp_lid<X>_ra<Y>

Set this value to the corresponding CPA group.

srtype_lid<X>_ra<Y>

Set this value to the appropriate CHI protocol version, either CHI-B, CHI-C, or CHI-D.

Local RN-F LDIDs must have the remote_ra<ldid> and cpa_en_ra<ldid> bits set to 0 and the corresponding CPA group ID set to 0.

2. Set each bit that must be used to hash snoop traffic across CXGs to 1 in the addr_mask field of the por_hnf_cml_port_aggr_grp<m>_addr_mask register. Set each bit that must be masked from the hash function to 0.

If CPA mode is enabled, HN-F SAM hashes PA bits[51:6] to distribute snoop traffic between CXGs. The PA is compared against the addr_mask field of the cml_port_aggr_grp<m>_addr_mask_register. Any PA bit that corresponds to a bit with a 0 value in the mask register is masked from hashing.

3. Program the characteristics of the CXGs for each CPAG by completing the following steps:
 - a. Program the number of CXGs in each CPAG by setting the num_cxg_pag<n> field in the por_hnf_cml_port_aggr_ctrl_reg register.
 - b. Program the CHI node ID of each CXG into the pag_tgid<n> fields in por_hnf_cml_port_aggr_grp_reg0 and por_hnf_cml_port_aggr_grp_reg1.

This programming is used by all regions that have CPA enabled.

4. Repeat the preceding steps for all HN-F SAMs in the chip.

Enable CCIX port-to-port forwarding

Use this procedure to enable port-to-port forwarding for your CML configuration. This programming must be completed for each CCIX port in the configuration. Port-to-Port forwarding can only be enabled in CXG blocks.

Each CXG block that is present on CMN-700 is a CCIX port and is assigned a unique logical ID when Socrates builds the CMN-700 configuration. This logical ID is known as the PortID of the CCIX port. You can determine the PortID for a CCIX port by reading the logical ID field (bits[47:32]) of por_cxg_ra_node_info register for the corresponding CXRA.

Enabling port-to-port forwarding for a CML configuration involves PortID assignment and configuration of registers that provide an AgentID to PortID LUT.

The following constraints apply to this process:

- PortIDs are only unique within a chip. In other words, CCIX ports on separate chips might have the same PortID.
- From a given port, all the TLP traffic targeting the same chip must go through the same target port. Failure to follow this constraint violates TLP to TLP ordering guarantees.
- Port credits are evenly distributed between all the communicating ports. Therefore, port_fwd_en must only be set for ports that the current port can receive forwarded traffic from.

Procedure

1. Set the port_fwd_en field in the por_cxla_portfwd_ctl register.

You must set this field for all CCIX ports that the current port can communicate with. The port_fwd_en field is a 16-bit vector where each bit represents the PortID on CMN-700.

2. Ensure that AgentID to PortID programming is done as described in [Program AgentID to PortID sets on page 3-1263](#).

Next Steps

To check the status of the port-to-port link, read the following register bits:

- port_fwd_req field in the por_cxla_portfwd_ctl register
- port_fwd_ack field in the por_cxla_portfwd_status register

The following bit combinations determine the status of the port-to-port link:

Disabled	port_fwd_req = 0 and port_fwd_ack = 0
Activate	port_fwd_req = 1 and port_fwd_ack = 0
Run	port_fwd_req = 1 and port_fwd_ack = 1
Deactivate	port_fwd_req = 0 and port_fwd_ack = 1

CCLA to CCLA Direct connect Mode

CML-CCG provides an option to directly connect CXS interface from CCLA on one CMN-700 to CXS interface of the other, without going through lower link layer and PHY controller IP. This mode is expected to be used in simulation environments for quick system bringup and must not be used when external controller IP is present.

The following steps must be followed to enable this correct connect mode.

1. Must set “ull_to_ull_en“ bit on both sides in CCLA’s por_ccla_ull_ctl register
2. After “ull_to_ull_en” is set on both sides (step#1) then set “send_vd_init “ bit on both sides in CCLA’s por_ccla_ull_ctl register

The following steps must be followed to quiesce and bring down the link (at the end)

Procedure

1. Clear “send_vd_init “ bit on both sides in CCLA’s por_ccla_ull_ctl register
2. Clear “ull_to_ull_en“ bit on both sides in CCLA’s por_ccla_ull_ctl register

See [por_ccla_ull_ctl](#) on page 3-558 and [por_ccla_ull_status](#) on page 3-560 for more details.

3.5.4 Program CMN-700 CML system at runtime

Use this procedure to program a CMN-700 CML system at runtime.

Procedure

1. Bring up CCIX protocol link.
For more information, see [3.5.5 Establish protocol link up between CXG and remote CCIX link on page 3-1269](#).
2. Add a CCIX protocol link in system coherency and DVM domains.
For more information, see [3.5.7 CCIX entry and exit protocol links from coherency domains and DVM domains](#) on page 3-1270.

Note

If CXSA mode is enabled, this programming is not necessary.

3. Program the remote address range and corresponding CXRA node ID for each remote memory region in RN SAM present in CMN-700 RN-F, RN-I, and RN-D.

RN SAM must not be programmed to target CXRA when enabled for CXSA mode.

Note

If the software can guarantee that there is no traffic to the remote address range until CCIX-related initial programming is complete and CCIX protocol links are up, then this programming should be done when programming RN SAMs with local address map.

3.5.5

Establish protocol link up between CXG and remote CCIX link

Use the following procedure to link up a CXG with a remote CCIX link that the CXG can communicate with.

A CCIX protocol link can be established between each CXG in CMN-700 and a corresponding remote CCIX link that the CXG can communicate with. The term *link* is used here to refer to a CCIX protocol link. Multiple CCIX links can be set up simultaneously by extending this sequence for each link.

Procedure

1. Poll the Lnk<X>_link_en bit in both por_cxg_ra_cxpctl_link<X>_ctl and por_cxg_ha_cxpctl_link<X>_ctl to ensure that the link is enabled.
If the link is enabled, then communication on the link can be established.
2. Ensure that the link is down and can accept a new link up request by polling the following bits:
 - a. Poll the Lnk<X>_link_up bits in the por_cxg_ra_cxpctl_link<X>_ctl and por_cxg_ha_cxpctl_link<X>_ctl registers to ensure that they are clear.
 - b. Poll the Lnk<X>_link_down bits in the por_cxg_ra_cxpctl_link<X>_status and por_cxg_ha_cxpctl_link<X>_status registers to ensure that they are set.
 - c. Poll the Lnk<X>_link_ack bits in the por_cxg_ra_cxpctl_link<X>_status and por_cxg_ha_cxpctl_link<X>_status registers to ensure that they are clear.
3. Set the Lnk<X>_link_req bits in the por_cxg_ra_cxpctl_link<X>_ctl and por_cxg_ha_cxpctl_link<X>_ctl registers.
Setting this bit generates a request to link up, which in turn brings up the link.
4. Poll the following bits in the por_cxg_ra_cxpctl_link<X>_status and por_cxg_ha_cxpctl_link<X>_status registers to ensure that the link up request is accepted:
 - Lnk<X>_link_ack
 - Lnk<X>_link_down

The hardware acknowledges a link up request by setting the Lnk<X>_link_ack bits and then clearing Lnk<X>_link_down bits in CXRA and CXHA.

Results: Hardware acknowledgment of link up means that both sides are ready to receive and grant CCIX protocol credits.

5. Set the Lnk<X>_link_up bit in the por_cxg_ra_cxpctl_link<X>_ctl and por_cxg_ha_cxpctl_link<X>_ctl registers to instruct both sides to start granting credits.

Link<X> is now up. Both sides can now exchange CCIX protocol credits and protocol messages.

3.5.6

Link down CCIX protocol link between CXG and remote CCIX link

Use this procedure to deactivate a CCIX protocol link between a CMN-700 CXG and a remote CCIX link.

You must follow a specific process to bring down a CCIX protocol link between each CMN-700 CXG and the corresponding remote CCIX link that the CXG can communicate with. The term *link* is used here to refer to a CCIX protocol link. Multiple CCIX protocol links can be brought down at the same time by extending this sequence for each link.

CMN-700 CXGs contain CXRA, CXHA, and CXLA nodes.

Prerequisites

Software must ensure that the following conditions are met before initiating a link down sequence:

- There are no outstanding transactions that require CCIX message transfers across the link for their completion. This condition includes GIC-D in SMP mode.
- The protocol agents on both sides of the link are configured to not initiate new transactions across the link. For CMN-700:

1. Poll the `lnk<X>_ot_cbkwr` bit in the `por_cxg_ra_cxprtcl_link<X>_status` register to make sure that it is cleared. This step ensures that there are no outstanding CopyBack requests targeting the link.
2. Take the link out of system coherency and DVM domains. For more information, see [3.5.7 CCIX entry and exit protocol links from coherency domains and DVM domains on page 3-1270](#).
- If CXSA mode is enabled, CCIX links must not be brought down until all HN-Fs communicating with this CXSA are in OFF state.

Procedure

1. Ensure that the link is up and can accept a new link down request by polling the following bits in the `por_cxg_ra_cxprtcl_link<X>_ctl` and `por_cxg_ha_cxprtcl_link<X>_ctl` registers:
 - `lnk<X>_link_up`
 - `lnk<X>_link_req`

These bits must be set before the link can accept a new link down request.
2. Clear the `lnk<X>_link_req` bits in the `por_cxg_ra_cxprtcl_link<X>_ctl` and `por_cxg_ha_cxprtcl_link<X>_ctl` registers to make a link down request.
3. Poll the `lnk<X>_link_ack` bit in the `por_cxg_ra_cxprtcl_link<X>_status` and `por_cxg_ha_cxprtcl_link<X>_status` registers to ensure that it is cleared.

The hardware acknowledges a link down request by clearing the `lnk<X>_link_ack`. After a link down request is accepted, each side must stop granting local CCIX protocol credits and start returning remote CCIX protocol credits.
4. Poll the `lnk<X>_link_down` bits in the `por_cxg_ra_cxprtcl_link<X>_status` and `por_cxg_ha_cxprtcl_link<X>_status` registers to ensure that each side has received all its protocol credits and is ready to go down.

The hardware sets the `lnk<X>_link_down` bits to convey that it is ready for the link to go down.
5. Clear the `lnk<X>_link_up` bits in the `por_cxg_ra_cxprtcl_link<X>_ctl` and `por_cxg_ha_cxprtcl_link<X>_ctl` registers to instruct both sides to deactivate the link.

Link<X> is now down. No protocol message or credit transfers must occur across the link.

3.5.7

CCIX entry and exit protocol links from coherency domains and DVM domains

CMN-700 CXG blocks, which contain CXRA, CXHA, and CXLA nodes, can establish a CCIX protocol link with a remote CCIX link. Each CXG has software-programmable bits to allow the CCIX protocol links to enter and exit the system coherency domains and DVM domains.

The CXHA has 2 bits which facilitate snoop coherency domain entry and exit requests and acknowledgment:

- The `lnk<X>_snooppdomain_req` bit of the `por_cxg_ha_cxprtcl_link<X>_ctl` in each CXHA, controls snoop coherency domain requests for the link.
- The `lnk<X>_snooppdomain_ack` bit of the `por_cxg_ha_cxprtcl_link<X>_status` register, provides acknowledgment and status of the snoop coherency domain requests for the link.

The CXRA has 2 bits which facilitate DVM domain entry and exit requests and acknowledgment:

- The `lnk<X>_dvmdomain_req` bit in the `por_cxg_ra_cxprtcl_link<X>_ctl` register in each CXRA, controls DVM domain requests for the link.
- The `lnk<X>_dvmdomain_ack` bit in the `por_cxg_ra_cxprtcl_link<X>_status` register, provides acknowledgment and status of the DVM domain requests for the link.

For more information, see [2.3.9 RN entry to and exit from Snoop and DVM domains on page 2-92](#).

Chapter 4

SLC memory system

This chapter describes the optional SLC memory system which is implemented by HN-Fs in the mesh.

It contains the following sections:

- [*4.1 About the SLC memory system* on page 4-1272.](#)
- [*4.2 SLC memory system components and configuration* on page 4-1274.](#)
- [*4.3 Error reporting and software-configured error injection* on page 4-1286.](#)
- [*4.4 Transaction handling in SLC memory system* on page 4-1287.](#)
- [*4.5 HN-F class-based resource allocation and arbitration* on page 4-1295.](#)

4.1 About the SLC memory system

The SLC memory system consists of the HN-F protocol nodes in CMN-700.

There is a configurable number of instances (1-128) of the HN-F. Each HN-F node or slice has the following features:

- 0KB, 128KB, 256KB, 512KB, 1MB, 2MB, 3MB, or 4MB of SLC Data RAM and Tag RAM.
- Combined *Point-of-Coherency* (PoC) and *Point-of-Serialization* (PoS).
- SF size of 1MB, 2MB, 4MB, 8MB when configured with 16 ways, or 16MB when configured to have 32 ways.
- SF Tag RAM configurable to have 16 or 32 ways

Each HN-F in CMN-700 is configured to manage a specific portion of the total address space. For each portion of the address, each HN-F:

- Can cache data in SLC
- Manages PoC and PoS functionality for ordering and coherency
- Tracks RN-F caching in the SF

Each HN-F in CMN-700 is configured to do in-line ECC Check/Correct in the SLC/SF RAM access pipeline.

The SLC memory system has the following features:

- *Physically Indexed and Physically Tagged* (PIPT)
- Coherency granule is a fixed length of 64B. SLC line size is a fixed length of 64B
- Both SLC 16-way set-associative for 512MB, 1MB, 2MB, and 4MB sizes. 12-way for 3MB SLC configurations.
- Optionally, CMN-700 supports an *enhanced LRU* (eLRU) cache replacement policy that can be enabled by setting a bit in the configuration register. eLRU is Dynamic Biased Replacement Policy. 2 bits per set/way are used to track and predict how soon a cache line is expected to be used again. This information is dynamically adjusted based on a few reference sets. By default the SLC and SF victim selection policy is:
 - Pseudo random if all ways are valid
 - If there is invalid way, it is not necessary to select a victim
 - Victim selection is needed only if all ways are taken
- SLC and SF arrays:
 - Supports one-cycle, two-cycle, or three-cycle non-pipelined tag array
 - Supports two-cycle or three-cycle non-pipelined data array
 - SLC Tag, SF Tag, and SLC Data arrays are single-ported, supporting one read or write access with no concurrency available
 - SLC Tag, SF Tag, and SLC Data arrays are ECC SECDED protected, with inline ECC checking and correction
- 16, 32, 64 or 128-entry address and data buffer, which is known as the *PoC Queue* (POCQ), to service:
 - All transactions from the CHI interface
 - SLC evictions to the memory controller
 - SF evictions and associated WriteBacks to the memory controller
- Fully configurable Clustering of RN-F's to support upto 512 Caching agents tracked in SF
- CMO propagation to SN-F or SBSX:
 - Implements improved PCMO flow that is introduced in CHI-D
 - Conditional CMO propagation to the memory controller to support external DRAM caches
 - HN-F must be explicitly programmed using the por_hnf_sam_sn_properties register of the HN-F SAM to allow such propagation to each SN-F
- Supports protocol flow control using programmable Classes:
 - POCQ resources are allocated or rejected for protocol retry according to the class
 - POCQ resources are watermarked for different classes with user-configurable options

- Class based static grantee selection for CHI architecture credit return
- For more information, please refer to [4.5 HN-F class-based resource allocation and arbitration on page 4-1295](#)
- Class based request selection to memory controller
- Supports allocation in the SLC from Snoop intervention. This feature enables data sharing through the SLC for multiple sharers.
- SLC state includes a caching LDID to detect dynamic read sharing
- Configurable 34-bit, 44-bit, 48-bit, or 52-bit *Physical Address* (PA) support
- PoC and PoS for all Snoopable and Non-snoopable, and Cacheable and Non-cacheable address space
- Supports ECC scrubbing for single-bit ECC errors on SF and SLC Tag RAMs
- Software-controlled error injection support to enable testing of software error handler routine
- Power management states to support:
 - Full powerdown of the SLC and SF. HN-F only mode when both SLC and SF are powered down
 - Half the SLC ways powered down
 - Retention for SLC and SF
 - SLC full powerdown with SF on, when in SF only mode
- Arm TrustZone® technology support in SLC and SF
- Software-configurable (one, two, four, eight, or 12 ways) Memory Region locking support in the SLC
- Software-configurable (one, two, four, eight, or 12 ways) OCM support in the SLC
 - OCM memory does not need any physical memory backing
- Supports CHI enhancements for:
 - *Direct Cache Transfer* (DCT)
 - *Direct Memory Transfer* (DMT). DMT not supported with 128b SBSX configurations
 - Cache Stashing
 - Atomics support
 - Data Poison
 - Data Parity (Data Check)
 - Trace Tag
- Invisible SLC support:
 - CMN-700 HN-F implements an invisible cache. All accesses (cacheable, non-cacheable, and Device types) are checked against the SLC and SF. The SLC cannot be cleaned and invalidated (flushed) by software using Arm architecture set/way operations. Software specific to CMN-700 would instead be required to flush the SLC, as described in this TRM. Invisible SLC support eliminates the need to perform SLC flushes for software context switches from cacheable to non-cacheable.
- Supports up to two memory-region-based SN targets and one, three, five, or six SN-F address hashing
- Supports CHI-D MPAM
- Supports CHI-E Memory Tagging Extensions (MTE)

4.2 SLC memory system components and configuration

CMN-700 *System Level Cache* (SLC) is a distributed, mostly exclusive last-level cache that is implemented within the HN-F node.

When a sharing pattern is detected between RN-F clusters, the SLC is optimized to eliminate redundancy for private data lines from the RN-F. The SLC also enables redundancy, or pseudo-inclusion. CMN-700 SLC also acts as DRAM cache for I/O coherent agents, that is, RN-Is. The SLC enables RN-Is to allocate or not allocate, according to the usage model.

The SF works with the SLC to track coherent lines that are present in the RN-F caches. The SF is fully inclusive of all the lines present in the RN-F caches. SF eviction invalidates the lines from RN-F caches to maintain this inclusion.

Normally, a particular coherent cache line is present only in the system level cache or SF except when the line is shared between RN-F clusters. In the shared case, the line can be present in both the SLC and the SF.

This section contains the following subsections:

- [4.2.1 HN-F configurable options on page 4-1274](#).
- [4.2.2 Snoop connectivity and control on page 4-1275](#).
- [4.2.3 TrustZone technology support on page 4-1275](#).
- [4.2.4 HN-F SAM configuration by SN type on page 4-1275](#).
- [4.2.5 Hardware-based cache flush engine on page 4-1276](#).
- [4.2.6 Software configurable memory region locking on page 4-1278](#).
- [4.2.7 Software-configurable On-Chip Memory on page 4-1279](#).
- [4.2.8 Source-based SLC cache partitioning on page 4-1280](#).
- [4.2.9 Way-based SLC cache partitioning on page 4-1280](#).
- [4.2.10 RN-F tracking in the SF on page 4-1282](#).
- [4.2.11 Non-clustered and clustered mode for SF RN-F tracking on page 4-1283](#).
- [4.2.12 Configuring non-clustered RN-F tracking in HN-F SF on page 4-1283](#).
- [4.2.13 Configuring clustered mode for SF tracking on page 4-1284](#).
- [4.2.14 Identifying clusters and individual devices in clustered mode on page 4-1284](#).

4.2.1 HN-F configurable options

The HN-F can be configured in several ways.

The HN-F has the following configurable parameters:

- SLC size of 0KB, 128KB, 256KB, 512KB, 1MB, 2MB, 3MB, or 4MB.
- SF size of 512KB, 1MB, 2MB, 4MB, 8MB, or 16MB
- SF 16 or 32 way set associate
- 16, 32, 64, or 128 POCQ entries.
- One-cycle, two-cycle, or three-cycle Tag RAM arrays. For a given configuration, both SLC Tag and SF Tag have the same latency.
- Two-cycle or three-cycle Data RAMs, data, and SF array RAMs. All Data RAMs have the same latency.
- RN-F clustered mode and number of RN-Fs that are in each cluster using `SF_MAX_RNF_PER_CLUSTER`. `SF_MAX_RNF_PER_CLUSTER` supports values of 1, 2, 4, and 8.
- Number of extra bits in the SF RN-F tracking vector to allow for hybrid clustering of RNs using `SF_RN_ADD_VECTOR_WIDTH`. `SF_RN_ADD_VECTOR_WIDTH` has a minimum value of 0 and a maximum value of 127. The total number of bits in the SF RN vector must not exceed 128 bits. The total can be calculated as $(\text{NUM_LOCAL_RNF} + \text{NUM_REMOTE_RNF}) / \text{SF_NUM_RNF_PER_CLUSTER} + \text{SF_RN_ADD_VECTOR_WIDTH}$.

The HN-F has the following fixed parameters:

- HN-F CHI interface data-VC (DAT) width of 256 bits.

4.2.2 Snoop connectivity and control

Each HN-F can send three types of snoop.

The available types of snoop request are:

- Directed, to one RN-F.
- Multicast, to more than one but not all.
- Multicast, to more than one but not all. If SF clustered mode is enabled, multicast snoops might be more common.
- Broadcast, to all RN-Fs. When SF clustering is enabled, HN-F may utilize CHI-E SnpQuery opcode. It can be sent to an RN-F that requests MakeReadUnique to HN-F. This helps HN-F determine the cache line presence RN-F at the time of processing the transaction.

4.2.3 TrustZone technology support

The HN-F supports TrustZone technology by treating the Non-secure bit from a request as part of the address.

TrustZone enables the HN-F to treat Secure and Non-secure as two different areas of the memory space:

- The NS bit is stored in the SLC and SF tags.
- Snoops also propagate the NS bit as part of the message.
- Any request to the memory controller also propagates the NS bit.

4.2.4 HN-F SAM configuration by SN type

CMN-700 supports multiple SN types. You must program the HN-F SAM according to the types of SN that the HN-F targets.

You can configure CMN-700 to use the following SN types:

- SBSX.
- CXSA.
- CHI-C, CHI-D, or CHI-E SN-F.
- MTSX.

To configure SN target types in the HN-F SAM, program the por_hnf_sam_sn_properties register. For the description of this register, see [por_hnf_sam_sn_properties on page 3-1139](#).

The following table shows the bit values that you must program for each SN type.

————— Note —————

The value of <x> describes the specific SN target.

Table 4-1 por_hnf_sam_sn_properties SN type values

SN type	<x>_sn_is_chic	<x>_sn_is_chie	<x>_sn_pcmossep_conv_to_pcmo
CHI-C SN-F	0b1	0b0	0b1
CHI-D SN-F	0b0	0b0	0b0
CHI-E SN-F	0b0	0b1	0b0
SBSX or MTSX	0b0	0b1	0b0

Note

- If the AXI memory controller does not support WR+PCMO, then HN-F must be programmed for this SN in CHI-D mode.
 - If the AXI memory controller does not support PCMO on AW channel, then HN-F must be programmed as CHI-C SN mode.
 - If the system uses CHI-E GrpIDExtn bits and the SN-F is CHI-D, then the `sn<x>_pcmo_conv_to_pcmo` for HN-F must be set to `0b1`.
 - If all SN types in the system do not support MTE, `hnf_mte_mode_dis` must be set to 1.
-

4.2.5 Hardware-based cache flush engine

The HN-F supports a hardware-based cache flush engine mechanism to flush the SF and SLC. The flush engine ensures that all cache lines in the lower and upper range are flushed from the CMN-700 SF and SLC.

Various *Address Based Flush* (ABF) configuration registers per HN-F instance support the cache flush engine:

<code>por_hnf_abf_lo_addr</code>	ABF lower range address.
<code>por_hnf_abf_hi_addr</code>	ABF upper range address.
<code>por_hnf_abf_pr</code>	ABF Policy Register. Triggers flush start, indicates flush operation type.
<code>por_hnf_abf_sr</code>	ABF Status Register. Indicates flush completion and other status information.

The flush engine ensures that all cache lines in the lower and upper range are flushed from the CMN-700 SF and SLC. When all cache lines within this range are flushed, a bit in the `por_hnf_abf_sr` register is set indicating that the flush engine has completed. If enabled, an interrupt (**INTREQPPU**) is then sent.

Note

Interrupt indication and complete bit in the `por_hnf_abr_sr` registers are set regardless of normal completion or abort condition. To determine if a flush request completed normally or aborted, check the error bits in the `por_hnf_abf_sr` register.

To complete the flush sequence, the HN-F carries out the following steps:

1. Flush CMN-700 SFs. This operation flushes the lines in the lower-level caches. Lower-level write-backs go to memory and are not allocated to the CMN-700 SLC.
2. Flush the CMN-700 SLC.
3. On completion of the flush:
 - a. The HN-F sets the status bit in the `por_hnf_abf_sr` register when the flush is complete for that HN-F. If there are error conditions, they are also set in the `por_hnf_abf_sr` register. This register is cleared when next ABF request starts.
 - b. The HN-F sends a completion message to the global Power/Clock/Reset unit, and an optional **INTREQPPU** is asserted when all HN-F instances have completed the flush.

ABF requests are processed in parallel to other ongoing requests from RNs. If an ABF request and another ongoing request target the same address, then no ordering or coherency guarantee is provided. Power management transition requests have higher precedence than ABF requests. An ABF request is only supported when the Power management state is in FAM, HAM, or SFONLY mode and the retention state is IDLE or RETENTION (not transitionary). While ABF is in progress, any update to the *Power Policy Register* (PWPR) causes the ABF state machine to abort and the Power management request proceeds.

By default, the flush engine writes back any modified data to memory before invalidating the cache line from internal caches. Two more configuration modes are provided for user flexibility. Therefore, the flush engine has the following three modes of operation:

CleanInvalid Write back and invalidate (default).

MakeInvalid In this mode, modified data is not written back to memory.

CleanShare In this mode, modified data is written back to memory but clean data remain in internal caches.

If *On Chip Memory* (OCM) is enabled and the address range overlaps with the ABF range, OCM behavior supersedes ABF. For SLC, this condition means that for CleanInvalid and CleanShare modes, no action is taken. For MakeInvalid, there is no difference in behavior regardless of whether the address is in the OCM range or not. For SF flush, the behavior is the same between an OCM address match and not.

The following tables show a summary for SF and SLC caches for all three modes.

Table 4-2 SF cache operation

SF state		Hit			Miss	
ABF mode	SNP type to RN-F	Change SF state?		SNP to RN-F	Change SF state?	
CleanInvalid	CleanInvalid	Yes		N/A	No	
MakeInvalid	MakeInvalid	Yes		N/A	No	
CleanShared	CleanShared	Yes		N/A	No	

Table 4-3 SLC cache operation

SLC state		Modified		Exclusive or Shared		Invalid	
ABF mode	OCM match?	Evict line?	Change L3 state? (Final state)	Evict line?	Change L3 state? (Final state)	Evict line?	Change L3 state (Final state)
CleanInvalid	No	Yes	Yes (I)	No	Yes (I)	No	No (I)
	Yes	No	No (M)	No	No (ES)	No	No (I)
MakeInvalid	No	No	Yes (I)	No	Yes (I)	No	No (I)
	Yes	No	Yes (I)	No	Yes (I)	No	No (I)
CleanShared	No	Yes	Yes (E)	No	No (ES)	No	No (I)
	Yes	No	No (M)	No	No (ES)	No	No (I)

Note

The following assumptions are made:

- To ensure coherency and ordering is maintained, RNs should not access a cache line within the flush range while ABF is in progress.
- Address ranges and trigger must be programmed for each HN-F in the SCG.
- Do not change ABF-related configuration register bits when the abf_enable bit of the por_hnf_abf_pr register is set, until the abf_complete bit of the por_hnf_abf_sr register indicates flush is done.
- HN-F must be in one of the three operational modes (FAM, HAM, SFONLY). When flush starts, any update to the PWPR causes ABF to abort.
- SF must be enabled for the flush engine to operate. If SF is disabled, the flush engine aborts and indicates an error status in the por_hnf_abf_sr register.
- When ABF completes, check the por_hnf_abf_sr to ensure ABF completed without any errors. If ABF aborted for any reasons, then the por_hnf_abf_sr indicates that the flush was aborted.

4.2.6

Software configurable memory region locking

The HN-F supports variable size memory regions that can be locked in the system level cache with way reservation.

These variable size memory regions ensure that locked lines are not evicted from the SLC, and any access to those lines is guaranteed to hit in the SLC. The variable memory region can be one of the following sizes:

- 0.5MB
- 1MB
- 2MB
- 4MB
- 8MB

These variable size memory regions ensure that locked lines are not evicted from the SLC, and any access to those lines is guaranteed to hit in the SLC. This guarantee applies to normal-memory cacheable and allocatable requests as described in CHI specifications. Non-allocating type requests may still be pushed to memory. The variable memory region is calculated as a factor of the total SLC size and number of ways that are locked. For example, If the SLC is built with 16 ways, then way locking of 1, 2, 4, 8 or 12 yields 1/16, 2/16, 4/16, 8/16 or 12/16 of the SLC size respectively.

Software uses the following mechanism to program the HN-F configuration registers to enable region locking:

- The `hnf_slc_lock_ways` register specifies the total number of locked HN-F system level cache ways. This register can have a value of 1, 2, 4, 8, or 12.
- The following region base registers specify the base address of the region that is using locked ways:
 - `hnf_slc_lock_base0` register.
 - `hnf_slc_lock_base1` register.
 - `hnf_slc_lock_base2` register.
 - `hnf_slc_lock_base3` register.
- A combination of the total SLC size, `hnf_slc_lock_ways` register, and the `hnf_slc_lock_base0` register to `hnf_slc_lock_base3` register defines the following:
 - The total amount of cache that is locked, calculated as follows:

$$\frac{\text{Total SLC size} \times \text{Number of locked ways}}{16}$$

Figure 4-1 Total cache locked equation

- Ways are locked beginning with way 0 and then in ascending order.
- The number of valid regions and exactly which regions, and therefore which of the `hnf_slc_lock_base0` to `hnf_slc_lock_base3` registers, are valid and included in the HN-F way allocation.
- The exact location, size, and alignment requirement of each region.
- The region alignment is identical to the region size, for example:
 - A 0.5MB region is aligned to any 0.5MB boundary.
 - A 4MB region is aligned to any 4MB boundary.
- The size and alignment requirement is enforced in hardware, to prevent any errors in software.
- Regions can be disjointed or contiguous, to create a larger single region.
- All valid regions use all locked ways. There is no application-level way segregation.
- The HN-F must be in the FAM power state. Memory Region Locking is not supported in other CMN-700 power states.

————— Note —————

The locked regions do not comprehend Secure as opposed to Non-secure memory regions, so if aliasing is performed between Secure and Non-secure regions, overlocking can occur.

The following tables specify various combinations of region size and the number of locked ways that software must program using the hnf_slc_lock_ways register and the hnf_slc_lock_base0 register to hnf_slc_lock_base3 register.

Table 4-4 SLC Region Lock sizes

SLC size	Number of locked ways	Total locked region size	Locked ways	Number of ways per region	Region 0	Region 1	Region 2	Region 3
8MB	1	0.5MB	0	1	0.5MB	-	-	-
8MB	2	1MB	0-1	1, 1	0.5MB	0.5MB	-	-
8MB	4	2MB	0-3	1, 1, 1, 1	0.5MB	0.5MB	0.5MB	0.5MB
8MB	8	4MB	0-7	2, 2, 2, 2	1MB	1MB	1MB	1MB
8MB	12	6MB	0-11	2, 2, 4, 4	1MB	1MB	2MB	2MB

Table 4-5 Settings for hnf_slc_lock_baseX

Region size	Valid bits
0.5MB	[PA_WIDTH-1:19]
1MB	[PA_WIDTH-1:20]
2MB	[PA_WIDTH-1:21]
4MB	[PA_WIDTH-1:22]
8MB	[PA_WIDTH-1:23]

4.2.7 Software-configurable On-Chip Memory

The CMN-700 HN-F supports software configurable *On-Chip Memory* (OCM) which allows for the creation of systems without physical DDR memory. It also allows a system to use SLC as scratchpad memory.

In OCM mode, the HN-F does not send requests to the SN-F. To enable OCM, the following requirements must be met:

- The HN-F must be in the FAM power state. Other CMN-700 power states are not supported in OCM mode.
- All OCM ways must be same across all HN-Fs in a system cache group.
- OCM mode must be enabled before any non-config accesses are sent to HN-F.

In OCM mode, the following CMOs terminate in the SLC:

- CleanInvalid and CleanShared CMOs terminate in the SLC without performing a WriteBack to the SN-F.
- CleanInvalid and CleanShared CMOs terminate in the SLC without invalidation or performing a WriteBack to the SN-F.
- MakeInvalid invalidates the cache line in SLC, and can be used to invalidate the OCM region.

OCM mode can be enabled by programming the hnf_ocm_en bit in the por_hnf_cfg_ctl register. If the hnf_ocm_allways_en bit is set to 1, then all transactions targeting the HN-Fs have OCM behavior. The OCM region must be contiguous and aligned to the total SLC size of the configuration when the

hnf_ocm_allways_en is set to 1. If the hnf_ocm_allways_en bit is 0, region locking registers define the OCM regions. For more information about these region locking registers, see [4.2.6 Software configurable memory region locking on page 4-1278](#).

————— Note ————

Region locking registers do not explicitly control Secure and Non-secure memory regions. Therefore combined Secure and Non-secure memory regions should not exceed the total SLC size that is locked for OCM.

4.2.8 Source-based SLC cache partitioning

HN-F supports a feature to lock SLC ways for requesting nodes RN-F, RN-I, and RN-D.

This feature is an extension of the address-based way locking but the locked ways are based on the requestors instead of the programmed address. CMN-700 supports programming of the number of ways that can be locked, RN devices for which these ways are locked and allocation policies in each HN-F. With this feature enabled, the locked SLC ways are only available to the programmed RNs for any new cache line allocations.

Source-based way locking feature can be enabled by programming the por_hnf_rn_region_lock.rn_region_lock_en bit to 0b1 in each HN-F instance. The requesting nodes, for which these ways are to be locked must also be explicitly enabled in the por_hnf_rn*region_vec registers. The requesting nodes are individually identified using the logical IDs or cluster IDs. See [4.2.13 Configuring clustered mode for SF tracking on page 4-1284](#). Each requesting node type RN-F, RN-I, and RN-D have different registers and are uniquely identified in CMN-700 system using logical IDs. The number of ways that are locked are programmed in the por_hnf_slc_lock_ways.ways field.

The region locking feature has two allocation modes:

- Allocating new cache lines for matching RNs only in the locked ways. By doing so, the matching RNs are restricted to allocate to the locked partition only. This mode can be enabled by setting por_hnf_rn_region_lock.rn_pick_locked_ways_only bit to 0b1.
- Allocating new cache lines for matching RNs to one of the locked or unlocked ways. This mode is the default behavior. In this mode, the locked ways are restricted only to the matching RNs but the unlocked ways are accessible by all the RNs.

Source-based SLC cache partitioning is supported only when *Enhanced LRU* (eLRU) mode is used.

————— Note ————

The HN-F must be in the FAM power state. Source-based cache partitioning is not supported in other CMN-700 power states.

4.2.9 Way-based SLC cache partitioning

Each SLC cache instance can be partitioned into different regions. This partitioning allows each requesting node (RN-F, RN-I, RN-D) to allocate in one or more regions, each consisting of four consecutive ways.

Each region group has configuration registers that indicate which of the logical RN-F/RN-I/RN-D masters can allocate to the corresponding group of SLC ways. By default, all RNs can allocate all 16 ways, as the following tables show.

Table 4-6 Logical RN-F ID requesting node

Register	Address offset	Ways reserved	Default	Logical RN-F ID								
				63	62	61	60	----	3	2	1	0
por_hnf_slcway_partition0_rnf_vec	0xC48	[3:0]	{64'{1'b1}}									
por_hnf_slcway_partition1_rnf_vec	0xC50	[7:4]	{64'{1'b1}}									
por_hnf_slcway_partition2_rnf_vec	0xC58	[11:8]	{64'{1'b1}}									
por_hnf_slcway_partition3_rnf_vec	0xC60	[15:12]	{64'{1'b1}}									

Table 4-7 Logical RN-I ID requesting node

Register	Address offset	Ways reserved	Default	Logical RN-I ID								
				31	30	29	28	----	3	2	1	0
por_hnf_slcway_partition0_rni_vec	0xC68	[3:0]	{32'{1'b1}}									
por_hnf_slcway_partition1_rni_vec	0xC70	[7:4]	{32'{1'b1}}									
por_hnf_slcway_partition2_rni_vec	0xC78	[11:8]	{32'{1'b1}}									
por_hnf_slcway_partition3_rni_vec	0xC80	[15:12]	{32'{1'b1}}									

Table 4-8 Logical RN-D ID requesting node

Register	Address offset	Ways reserved	Default	Logical RN-D ID								
				31	30	29	28	----	3	2	1	0
por_hnf_slcway_partition0_rnd_vec	0xC88	[3:0]	{32'{1'b1}}									
por_hnf_slcway_partition1_rnd_vec	0xC90	[7:4]	{32'{1'b1}}									
por_hnf_slcway_partition2_rnd_vec	0xC98	[11:8]	{32'{1'b1}}									
por_hnf_slcway_partition3_rnd_vec	0xCA0	[15:12]	{32'{1'b1}}									

The registers in these tables are used to mask the ways that are available for an RN to allocate to at all times. A value of:

0b1 Indicates that the corresponding Logical RN ID or cluster ID can allocate in this region.

For more information about SF clustered mode and cluster IDs, see [4.2.13 Configuring clustered mode for SF tracking on page 4-1284](#).

0b0 Indicates that the corresponding Logical RN ID cannot allocate to this region.

To enable the way reservation only to a subset of RNs, the mask in the preceding registers must be programmed as the following example shows:

Example 4-1 Reserve ways 0-3 for RN-F {0-3}

-
1. Write $64'h000000000000000F$ to por_hnf_slcway_partition0_rnf_vec. This operation enables logical RN-F IDs 0, 1, 2, and 3 to allocate to ways 0-3. All other RN-F IDs (4-63) cannot allocate to these ways.
 2. Write $32'h0$ to por_hnf_slcway_partition0_rni_vec. This operation disables all 32 RN-Is from allocating to ways 0-3.
 3. Write $32'h0$ to por_hnf_slcway_partition0_rnd_vec. This operation disables all 32 RN-Ds from allocating to ways 0-3.
-

Note

The following conditions apply to this example:

- This feature cannot be used if region-based locking, source-based locking, or OCM is enabled.
- The region registers can be changed at runtime.
- When the way partitioning scheme is not being used, the preceding registers must be returned to the default values.
- Each RN should be configured to allocate to at least one partition. Setting the bit for a given RN to 0 in all partition registers defaults it to allocating in any partition.
- RN-I and RN-D each support a maximum logical ID of 32 in the partition mask register. In CML configurations, care must be taken to assign LDID to remote RN-I and RN-Ds above the local RN-I and RN-D logical IDs. This requirement ensures that SLC partitioning is honored correctly across all RN-Is and RN-Ds.
- RN-I and RN-D each support a maximum logical ID of 32 in the partition mask register.
- If SF clustered mode is enabled, HN-F uses the cluster ID instead of the full logical ID of the RNs. Therefore, all the RNs within a cluster can allocate to the locked ways.

Way-based SLC cache partitioning is supported only when *Enhanced LRU* (eLRU) mode is used.

Note

The HN-F must be in the FAM power state. Cache partitioning is not supported in other CMN-700 power states.

4.2.10 RN-F tracking in the SF

The CMN-700 HN-F SF tracks cache lines that come from specific RN-Fs using an RN-F vector index. Your CMN-700 configuration and the values of some configurable parameters determine the size of this index.

The maximum width of the vector index is 128 bits, and therefore it is limited to 128 entries.

The total width of the SF vector index is calculated using the values of various parameters, using the following calculation:

$$SF_TOTAL_WIDTH = SF_MIN_WIDTH + SF_RN_ADD_VECTOR_WIDTH$$

The configuration calculates SF_MIN_WIDTH, using the following calculation:

$$SF_MIN_WIDTH = \text{ceil}((NUM_LOCAL_RNF + NUM_REMOTE_RNF) / SF_NUM_RN_PER_CLUSTER)$$

The SF_MIN_WIDTH calculation ensures that the SF vector index can track all the local and remote RN-Fs in your configuration. However, you can also configure the SF vector index at build time to contain extra bits by using the SF_RN_ADD_VECTOR_WIDTH parameter. This parameter lets you add extra bits to the SF vector, up to a combined maximum SF_TOTAL_WIDTH of 128 bits.

You can choose between two modes for tracking RN-Fs in the SF, or you can use a mixture of the modes. For more information about these modes, see [4.2.11 Non-clustered and clustered mode for SF RN-F tracking on page 4-1283](#).

4.2.11 Non-clustered and clustered mode for SF RN-F tracking

The CMN-700 SF has two modes for tracking cache lines from RN-Fs: non-clustered and clustered mode. The mode that you use affects the number of RN-Fs that the SF can track.

Non-clustered mode

Each entry in the RN-F vector index is associated with a single RN-F. In this mode, CMN-700 is limited to a maximum of 128 RN-Fs. This limitation is because the SF can only track cache lines from a maximum of 128 RN-Fs. This limitation also applies to CML configurations.

Clustered mode

Each entry in the RN-F vector index is associated with a group of RN-Fs. You can group two, four, or eight RN-Fs into a cluster using this mode. This mode lets you increase the number of RN-Fs in a system up to a maximum of 512.

Both non-clustered and clustered mode use RN-F LDIDs to map RN-Fs to entries in the index. However, the way that the SF uses LDIDs to map RN-Fs to index entries differs between the two modes. In non-clustered mode, the whole LDID is used to map an RN-F to a single entry. In clustered mode, part of the LDID indicates which cluster group an RN-F belongs to and another part indicates the device within the group. In this case, the first part of the LDID is known as the cluster ID, and the second part is known as the device ID. Each bit in the SF vector represents the cluster ID of the RNs. For example, If there are four RN-Fs in a cluster, the SF is addressing all four RN-Fs. For any shared lines, HN-F snoops all four RN-Fs in the cluster.

For more information about the modes, including example configurations, see the following sections:

- [4.2.12 Configuring non-clustered RN-F tracking in HN-F SF on page 4-1283](#)
- [4.2.13 Configuring clustered mode for SF tracking on page 4-1284](#)
- [4.2.14 Identifying clusters and individual devices in clustered mode on page 4-1284](#)

4.2.12 Configuring non-clustered RN-F tracking in HN-F SF

Non-clustered mode is enabled using a configuration parameter. There are some constraints on how to use non-clustered mode in a CML configuration.

Non-clustered mode is one of two modes that can be used for RN-F tracking in the SF in a CMN-700 configuration. For more information about RN-F tracking, including an overview of the two modes, see [4.2.11 Non-clustered and clustered mode for SF RN-F tracking on page 4-1283](#). For more information about configuring clustered mode, see [4.2.13 Configuring clustered mode for SF tracking on page 4-1284](#).

Non-clustered mode is enabled when the `SF_MAX_RNF_PER_CLUSTER` parameter = 1. The LDID of each RN-F in the system is assigned to a single index entry in the SF RN-F vector index.

For example, the following table shows an example mapping between the RN-F vector index and the RN-F LDIDs in a configuration with 128 non-clustered RN-Fs.

Table 4-9 Example SF RN-F vector index for 128 non-clustered RN-F configuration

RN-F vector index value	RN-F LDID
0	LDID_0
1	LDID_1
2	LDID_2
...	...
127	LDID_127

In a CML configuration, all remote RNs must have LDIDs assigned to them after the local RN-F LDIDs have been assigned. For example, consider a system with 64 RN-Fs, divided into 16 local RN-Fs and 48 remote RN-Fs. In this system, LDIDs 0-15 are preassigned to the local RN-Fs. LDIDs 16-63 must then be assigned to the remote RN-Fs.

4.2.13 Configuring clustered mode for SF tracking

Initially, you configure clustered mode for SF tracking at build time. However, there are programming requirements and further information you must be aware of when configuring clustered mode.

You can enable clustered mode for RN-F tracking in the SF by setting the `SF_MAX_RNF_PER_CLUSTER` parameter value to a valid value that is >1. This parameter specifies the maximum number of RN-Fs that are in a cluster group, up to a maximum of eight.

If clustered mode is enabled, there must be at least two cluster groups. In other words, you cannot cluster all RN-Fs into a single cluster group.

If there is a single RN-F in the cluster, then it must always use the lowest device ID in that cluster. Consider a four-way clustering, where the device ID fields are `0b00`, `0b01`, `0b10`, and `0b11`. In this configuration, you cannot use device ID values of `0b01`, `0b10`, or `0b11` unless `0b00` is also in use.

Local and remote RN-Fs can be combined into the same cluster. Each cluster can contain any number of RN-Fs, up to the value of the `SF_MAX_RNF_PER_CLUSTER` value. For example, in a system with `SF_MAX_RNF_PER_CLUSTER = 4`, cluster 0 could contain four RN-Fs and cluster 1 could contain only two RN-Fs.

The size of the SF RN-F vector index is partly configurable, as described in [4.2.10 RN-F tracking in the SF on page 4-1282](#). You can use the `SF_RN_ADD_VECTOR_WIDTH` parameter to set up a mixed clustered and non-clustered tracking scheme. This type of scheme permits the SF to track some RN-Fs clustered in groups, and others individually.

For a configuration that uses clustered mode, you must ensure that each valid LDID must have the following attributes programmed appropriately in the `por_hnf_rn_cluster[0-127]_physid_reg[0-3]` registers:

- RN_Valid
- CPA enable
- CPA Group ID
- Remote
- CHI protocol supported
- RN NodeID

The LDID of local RNs can be reprogrammed according to the clustering requirements.

When choosing to use clustered mode, there are some further considerations to make:

- SLC partitioning and SLC way locking using source ID uses the cluster ID instead of the full LDID. Therefore, all RN-Fs in the cluster can use the partitioned ways for this source ID.
- Using clustered mode in the CMN-700 SF increases the likelihood of SF back invalidations. Although unique cache lines are still precisely tracked in the SF vector index, any lines that go into shared mode can cause SF pollution. SF pollution can occur because the evict operations from the clustered RN-Fs do not clear the SF entry.

4.2.14 Identifying clusters and individual devices in clustered mode

For the SF to track and identify RN-Fs within cluster groups, the RN-F LDID bits are divided into two separate components: the cluster ID and the device ID. The number of LDID bits that are used for these components depends on two properties of your configuration.

The first property is the number of RNs in the system that are addressable by the SF RN-F vector index. The width of the LDID scales up to a maximum size of 9 bits in a system with 512 RN-Fs.

The second property is the value of the `MAX_RNF_PER_CLUSTER` parameter.

Calculate the width of the device ID component of the LDID using the following equation:

$$\text{Device ID width} = \log_2(\text{MAX_RNF_PER_CLUSTER})$$

The device ID bits are mapped to the least significant bits of the LDID. The remaining bits of the LDID represent the cluster ID.

For example, the following table shows how LDID[8:0] is divided for different MAX_RNF_PER_CLUSTER values in a 512 RN-F configuration in clustered mode.

Table 4-10 Cluster ID and device ID LDID components for 512 RN-F configuration

MAX_RNF_PER_CLUSTER value	LDID ranges	Clustering components
4	LDID[8:2]	Cluster ID[6:0]
	LDID[1:0]	Device ID[1:0]
8	LDID[8:3]	Cluster ID[5:0]
	LDID[2:0]	Device ID[2:0]

It is important to take the cluster ID and device ID into account when assigning LDIDs to local and remote RN-Fs. Accounting for these values helps you to make sure that RN-Fs are grouped in the cluster as required.

Example cluster mode configurations

The following table shows an example configuration with 512 RN-Fs, a 128-bit RN-F vector index, and cluster groups of four RN-Fs.

Table 4-11 Example clustered mode configuration with four RN-Fs per cluster group

RN-F vector index value (cluster ID)	Device ID 0 (LDID)	Device ID 1 (LDID)	Device ID 2 (LDID)	Device ID 3 (LDID)
0	LDID_0	LDID_1	LDID_2	LDID_3
1	LDID_4	LDID_5	LDID_6	LDID_7
2	LDID_8	LDID_9	LDID_10	LDID_11
...
127	LDID_508	LDID_509	LDID_510	LDID_511

The following table shows an example configuration with 512 RN-Fs, a 64-bit RN-F vector index, and cluster groups of eight RN-Fs.

Table 4-12 Example clustered configuration with eight RN-Fs per cluster group

RN-F vector index value (cluster ID)	Device ID 0 (LDID)	Device ID 1 (LDID)	Device ID 2 (LDID)	Device ID 3 (LDID)	Device ID 4...6	Device ID 7 (LDID)
0	LDID_0	LDID_1	LDID_2	LDID_3	...	LDID_7
1	LDID_8	LDID_9	LDID_10	LDID_11	...	LDID_15
2	LDID_16	LDID_17	LDID_18	LDID_19	...	LDID_23
...
63	LDID_504	LDID_505	LDID_506	LDID_507	...	LDID_511

4.3 Error reporting and software-configured error injection

HN-F detects and reports several types of errors to the error block.

HN-F supports the following types of errors:

Correctable Errors

For example, single-bit ECC error detection and correction in the SLC Tag RAM, SF Tag RAM, and SLC Data RAM

Deferred Errors

For example, double-bit ECC error detection in SLC Data RAM

Uncorrectable Errors

For example, double-bit ECC errors in the SLC or SF Tag RAMs

If the DATACHECK_EN parameter is enabled, HN-F can also support Data parity error detection in the SLC Data RAM. These errors are logged as Deferred Errors.

For logging and reporting all error types, HN-F follows the procedures described in [2.8 Reliability, Availability, and Serviceability on page 2-175](#).

For information regarding the error source, see the ERRSRC field of [por_hnf_errmisc on page 3-1092](#).

4.3.1 Software-configurable error injection

The HN-F supports software-configurable error injection and reporting. This feature enables testing of the software error handler routine for SLC double-bit ECC data errors.

The HN-F configuration register for a particular logical thread enables configurable error injection and reporting.

Any Cacheable read for which the HN-F provides the data is defined as a system cache hit. If error injection and reporting are enabled, any system cache hit drives the slave error from the system cache pipe and a fault interrupt through the RAS control block for that read. This functionality emulates a double-bit ECC error in the SLC data RAM but does not pollute the SLC data RAM through the fill path.

Note

This mechanism is designed to mimic SLC data ECC errors for SLC hits. If enabled, this mechanism only causes an error to be logged and optionally an interrupt to be generated. SLC misses do not drive any slave errors or error interrupts. Error injection on SLC hits does not alter the Resp*, Poison, or Data fields in the DAT flit that is returned to the RN.

For more information about configuring error injection, see [por_hnf_err_inj on page 3-1094](#).

4.3.2 Software-configurable parity error injection

The HN-F supports software-configurable parity error injection.

This feature enables testing of the software error handler routine for parity error. For more information about enabling this feature, see [por_hnf_byte_par_err_inj on page 3-1095](#). This register specifies the byte lane from 0-31 in which a parity error is introduced. The memory data is uncorrupted with such injection, but the Data Check field of the DAT flit that is returned to an RN is altered.

4.4 Transaction handling in SLC memory system

The CMN-700 SLC memory system handles various types of CHI operations and transaction fields. The structure of the overall system and how each component is configured affects how these transactions are handled.

This section contains the following subsections:

- [4.4.1 Cache maintenance operations on page 4-1287](#).
- [4.4.2 Cacheable and Non-cacheable exclusives on page 4-1287](#).
- [4.4.3 DataSource handling on page 4-1288](#).
- [4.4.4 CMO and PCMO propagation from HN-F to SN-F/MTSX/SBSX on page 4-1288](#).
- [4.4.5 Memory System Performance Resource Partitioning and Monitoring on page 4-1290](#).
- [4.4.6 MTE support in HN-F on page 4-1294](#).

4.4.1 Cache maintenance operations

CMN-700 uses several CHI *Cache Maintenance Operations* (CMOs).

The following operations are supported:

- CleanInvalid.
- CleanShared.
- MakeInvalid.
- CleanSharedPersist.
- CleanSharedPersistSep.

These operations always look up the SLC and the SF, and take the following actions:

- Clean and invalidate the line if present in the SLC.
- If the CMO is Snoopable, the HN-F sends a snoop to the RN-F post SF lookup if necessary.
- If the cache line is modified in the SLC or in the cache of the RN-Fs, the HN-F initiates a memory controller WriteBack if necessary.

————— Note —————

If the CMO is MakeInvalid, there is no WriteBack to the memory controller.

The SF does not track RN-F coherence while the HN-F is in NOSFSLC state. Therefore, the RN-F caches must be flushed before transitioning from NOSFSLC to SFONLY, HAM, or FAM states.

The SF does not track RN-F coherence while the HN-F is in NOSFSLC state. Therefore, the RN-F caches must be flushed before transitioning from NOSFSLC to SFONLY, HAM, or FAM states.

CMN-700 supports propagation of CMO and PCMO requests for a given cache line to the memory controller. This feature ensures that the cache line has been written to the memory controller and any copies in the CMN-700 system have been removed. Conditional CMO and PCMO propagation to the memory controller also supports external DRAM caches. This feature can be enabled or disabled in each HN-F's por_hnf_sam_sn_properties register bits corresponding to each SN-F.

4.4.2 Cacheable and Non-cacheable exclusives

The HN-F supports PoC monitor functionality for Cacheable and Snoopable exclusive operations from the RN-Fs.

The Cacheable and Snoopable exclusive transactions are:

- ReadShared.
- ReadClean.
- CleanUnique.
- ReadPreferUnique

- MakeReadUnique
- ReadNotSharedDirty.

The HN-F also supports system monitor functionality for Non-cacheable exclusive support. For more information about exclusives, see the *AMBA® 5 CHI Architecture Specification*.

Note

Each HN-F in CMN-700 can support tracking of up to 512 logical processors for exclusive operations. In configurations with up to 64 RN-Fs, HN-F supports 64 exclusive monitors and in configurations with 64-144 RN-Fs, HN-F supports 144 exclusive monitors. In configurations above 144 RN-Fs, the total number of exclusive monitors is equivalent to the total number of local and remote RN-Fs, RN-Is, and RN-Ds, up to a maximum of 512 exclusive monitors. The system programmer must ensure that there are no more logical processors capable of concurrently sending exclusive operations than the number of exclusive monitors.

4.4.3 DataSource handling

CMN-700 populates the DataSource field of a CompData response to specify the source of the data.

DataSource information can be used:

- To determine the usefulness of a PrefetchTgt (Memory controller prefetch) transaction.
- To profile and debug software to evaluate and optimize data sharing patterns.

Table 4-13 DataSource encodings

Source of data	Message	Encoding
HN-I	Default (Non-memory source)	0b0000
RN-F	Peer processor cache within local cluster	0b0001
RN-F	Local cluster cache	0b0010
HN-F	<i>System Level Cache (SLC)</i>	0b0011
RN-F	Peer cluster cache	0b0100
Remote chip	Remote chip caches	0b0101
SN-F,SBSX, or MTSX	PrefetchTgt was useful.	0b0110
SN-F,SBSX, or MTSX	PrefetchTgt was not useful.	0b0111
RN-F	Local cluster cache, unused prefetch	0b1010
HN-F	<i>System Level Cache (SLC)</i> , unused prefetch	0b1011

Note

In MTSX, the indication of the usefulness of PrefetchTgt applies only to the data portion of the transaction. The indication does not apply to tags.

CMN-700 drives the DataSource value only when the source of the data is either HN-F or HN-I. For other data sources, CMN-700 acts as a conduit.

The encoding that is used by CMN-700 to indicate a data source is the same as the suggested value in the *AMBA® 5 CHI Architecture Specification*. Any deviation from the specified encodings might result in unexpected behavior.

4.4.4 CMO and PCMO propagation from HN-F to SN-F/MTSX/SBSX

CMN-700 supports propagation of CMO and PCMO requests for a given cache line to the memory controller. The completion point of these requests is programmable.

This feature ensures that the cache line has been written to the memory controller and any copies in the CMN-700 system have been removed. Conditional CMO and PCMO propagation to the memory controller also supports external DRAM caches. This feature can be enabled or disabled in the por_hnf_sam_sn_properties register bits for each HN-F corresponding to each SN-F. For SBSX with an AXI4 slave memory device, you must disable CMO and PCMO propagation must in HN-F.

The following HN-F SAM configuration register bits decide the CleanSharedPersist (PCMO) and CleanSharedPersistSep (PCMOSep) completion point:

- por_hnf_sam_sn_properties.pcmo_prop_dis
- por_hnf_cfg_ctl.hnf_pcmo_pop_en

You can program this behavior per SN. Deep is an attribute in the CHI-D request flit and applies to PCMO type requests (DPCMOs). HN-Fs use the Deep attribute and the HN-F SAM configuration bits to decide the PCMO request completion point. The following diagram shows the PCMO and PCMOSep request completion point decision process.

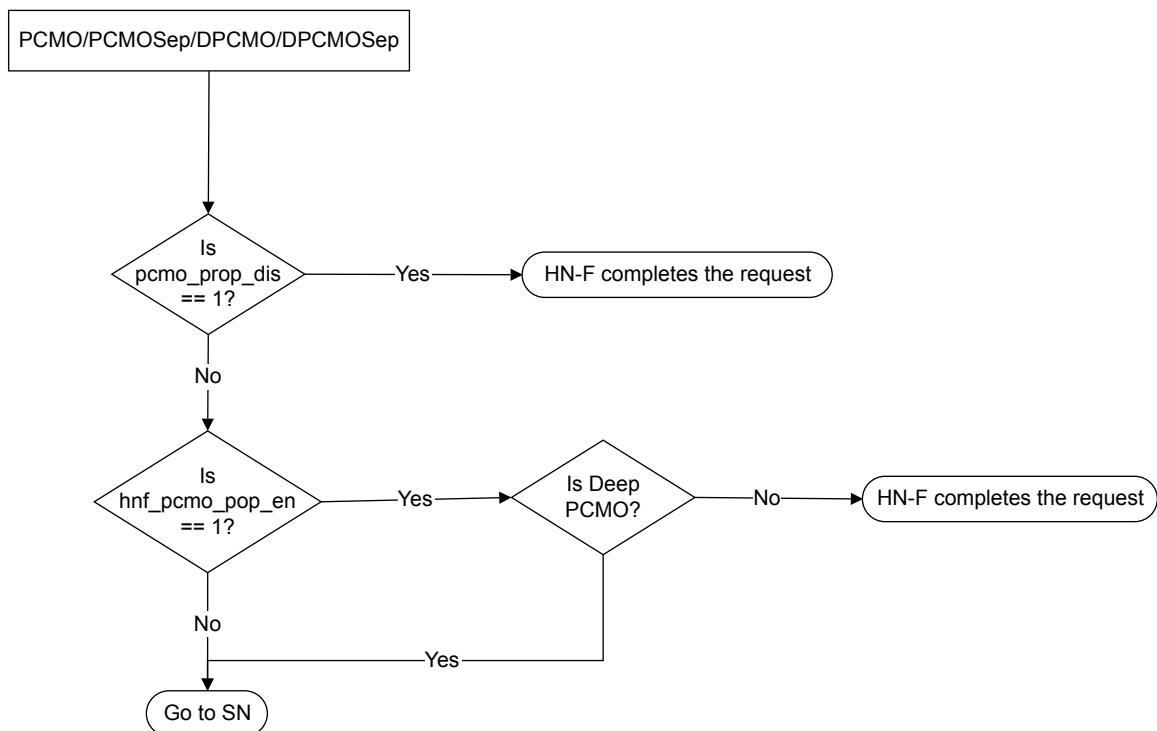


Figure 4-2 PCMO and PCMOSep completion point flow diagram

If an HN-F is programmed to propagate PCMO requests to an SN-F, and the SN-F does not support PCMOSep requests, you must set the por_hnf_sam_sn_properties.X_sn_pcmosep_conv_to_pcmo bit to **0b1**. This setting allows the HN-F to complete the request by converting PCMOSep type requests to PCMO. The HN-F generates a Persist response to the requestor on receiving a completion of PCMO response from SN-F.

You can also program HN-Fs to propagate CleanShared, CleanInvalid, and MakeInvalid CMOs to an SN-F. The following table shows how to program this behavior using the por_hnf_sam_sn_properties register.

Table 4-14 CMO propagation programming in HN-F SAM

HN-F SAM attribute (cmo_prop_en)	CMO completion point
0b0	HN-F
0b1	SN

See the following register descriptions for more information:

- [*por_hnf_sam_sn_properties* on page 3-1139](#)
- [*por_hnf_cfg_ctl* on page 3-1066](#)

4.4.5 Memory System Performance Resource Partitioning and Monitoring

CMN-700 supports *Memory System Performance Resource Partitioning and Monitoring* (MPAM). MPAM features enable software to optimize the use of memory resources and to monitor how those resources are used.

If MPAM is enabled, then an extra MPAM field is added to the REQ and SNP channels. This field must be stored and propagated to the downstream target.

For more information about MPAM, see the following documents:

- *AMBA® 5 CHI Architecture Specification*
- *Arm® Architecture Reference Manual Supplement Memory System Resource Partitioning and Monitoring (MPAM), for Armv8-A*

MPAM propagation

When MPAM is enabled, CMN-700 propagates the MPAM fields throughout the network. Various nodes are designed to propagate the MPAM field when processing requests.

The following node types propagate the MPAM field:

- MXP
- RN-I
- HN-F and HN-I
- CXRA and CXHA
- SBSX

Note

The AXMPAM_EN parameter controls if MPAM bits are stored and propagated on RN-I, HN-I, and SBSX bridges.

Note

If MPAM support is required in CMN-700, set the CHI_MPAM_ENABLE parameter.

When an HN-F node receives a request, it stores the details from the MPAM field. If the request misses in the SLC, then the HN-F must drive the stored MPAM values onto the outgoing request to the memory controller. The MPAM field contents are stored in the SLC_TAG array and propagated into requests that are sent to memory.

When MPAM is enabled, an extra register field, *_mpam_override_en, is added to the MXP. You can program this field to override the MPAM value of the RN-F request with the set value in the register. For more information, see [*por_mxp_p_0-5_mpam_override* on page 3-984](#).

When MPAM is enabled, the RN-I has the following extra components:

- Two extra signals on the AXI or ACE-Lite slave port, **ARMPAM[10:0]** and **AWMPAM[10:0]**.
- An extra register field, *_mpam_override_en, which you can program to override the request MPAM value with the set value in the register.

———— Note ————

If **AXMPAM_EN** = 0, then the MPAM override is active by default.

The CXRA node type has the following MPAM modes:

SMP mode

The CXRA passes the MPAM field on the USER field of the request. When snooped, the CXRA receives the MPAM field and passes it through the CHI SNP MPAM field.

Non-SMP mode

The CXRA drops the MPAM field that is received on the CHI request. The CXRA also does not receive MPAM field on CGL snoop in this mode.

CXSA mode

The CXRA passes the MPAM field on the USER field of the request, even though CXSA is in non-SMP mode. You can use a configuration bit to enable passing of MPAM attributes when in CXSA mode.

The CXHA node type has the following MPAM modes:

SMP mode

The CXHA receives MPAM fields through the USER field of the request and passes them through the CHI MPAM field. On incoming CHI transactions, the CXHA passes CHI MPAM values through on the USER field.

Non-SMP mode

The CXHA drops the MPAM values that it receives on CHI snoop. The CXHA does not receive the MPAM attributes on CGL request.

If **AXMPAM_EN** = 1, the HN-I and SBSX node types propagate **CHI.RXREQ.MPAM** onto the AXI pins. If **AXMPAM_EN** = 0, the HN-I and SBSX node types drive 0s onto AXI pins.

In CMN-700, MPAM support is applicable to the SLC.

MPAM configuration

CMN-700 provides several configuration parameters to configure MPAM features that are used by the interconnect. The MPAM feature ID register, **por_hnf_mpam_idr**, provides information on what MPAM features are supported in the design.

The number of partitions and the number of performance monitoring groups that are supported is configurable at build time. The default number of partitions is 64 for Non-secure partitions and 16 for Secure partitions. The default number of performance monitoring groups is two.

For more information about the configuration options, see [1.5.2 Mesh sizing and top-level configuration on page 1-21](#) and [1.5.3 Device placement and configuration on page 1-27](#).

For more information about supported MPAM features, see [por_hnf_mpam_idr on page 3-904](#).

For more information about the software-programmable MPAM override mechanism, see [Software-programmable MPAM override on page 4-1293](#).

Cache portion and capacity partitioning

MPAM support in the interconnect is for the CMN-700 SLC. MPAM is supported only with an *enhanced Least Recently Used* (eLRU) cache replacement policy.

The SLC supports cache portion partitioning that is based on the following masks:

MPAMCFG_CPB

The CPBM value for a request PARTID determines which cache portions a request can allocate.

MPAMCFG_CMAX

The CMAX value for a request PARTID determines the percentage of the SLC that a request can use.

Cache portion and capacity partitioning have the following features:

- Number of portions is the same as the number of ways in SLC.
- The hnf_mpam_ccap_idr_cmax_wd field of the por_hnf_mpam_ccap_idr register is set to 6. This setting provides granularity of 1.56% (1 / 2⁶) SLC for cache capacity partitioning.
- In HAM mode, portions 15:8 are aliased to portions 7:0. Cache capacity is adjusted for half the cache.
- CMN-700 supports address based locking (including OCM) with MPAM. Locked ways are not available for MPAM-based partitioning. Cache capacity is adjusted to account for locked ways.

————— Note ————

CMN-700 does not support source-based or way-based SLC partitioning with MPAM.

If using way locking with MPAM, you must program the lock registers first. Then, to determine the number of available portions, read the hnf_mpam_cpor_idr_cpbm_wd field of the por_hnf_mpam_cpor_idr register. Locked ways also reduce cache capacity.

HN-F MPAM counter values are not accurate when exiting retention state and can result in underflow conditions.

Cache capacity monitoring

This section describes cache capacity monitoring.

MPAM provides a mechanism for monitoring SLC usage:

- The HN-F MPAM_NUM_CSUMON parameter determines how many monitors are supported.
- The por_hnf_X_msmon_cfg_csu_flt and por_hnf_X_msmon_cfg_csu_ctl registers determine filter and control for each monitor.

————— Note ————

The interconnect has two banks for these registers, S and NS, denoted by X in the register or register field name.

The hnf_X_msmon_cfg_csu_ctl_capt_evnt field of the por_hnf_X_msmon_cfg_csu_ctl register supports external capture events 6 and 7:

- External capture event 6 is triggered using **PMUSAPSHOT** interface.
- External capture event 7 is triggered as described in the *Arm® Architecture Reference Manual Supplement Memory System Resource Partitioning and Monitoring (MPAM), for Armv8-A*.

————— Note ————

Multiple capture events cannot be triggered within 32 cycles of each other.

MPAM error logging and reporting

CMN-700 implements programmable registers that can enable, disable, and modify MPAM error logging and reporting behavior.

The MPAM error status register, por_hnf_X_mpam_esr, and the MPAM error control register, por_hnf_X_mpam_ecr, define MPAM-related error logging.

Note

The interconnect has two banks for these registers, S and NS, denoted by X in the register or register field name.

You can enable interrupt generation for MPAM-related errors by setting the hnf_X_mpam_ecr_inten field of the por_hnf_X_mpam_ecr register to 0b1. If interrupt generation is enabled, level-sensitive interrupts (**INTREQMPAMERRS** or **INTREQMPAMERRNS**) are triggered for defined error cases.

Note

MPAM error reporting has the following exceptions:

- When SLC size is 0K, no errors are detected or reported.
- REQ PARTID or PMG out of range errors are not detected or reported when:
 - HN-F is in SFONLY mode.
 - MPAM features are disabled (by configuring the auxiliary control register).

For more information about MPAM errors, see *Section 12.2, Error conditions in accessing memory-mapped registers* in the *Arm® Architecture Reference Manual Supplement Memory System Resource Partitioning and Monitoring (MPAM)*, for Armv8-A.

See the following register descriptions for more information:

- [por_hnf_ns_mpam_ecr](#) on page 3-918
- [por_hnf_ns_mpam_esr](#) on page 3-919
- [por_hnf_s_mpam_ecr](#) on page 3-741
- [por_hnf_s_mpam_esr](#) on page 3-742

Software-programmable MPAM override

CMN-700 has registers in various nodes to let software override the MPAM values generated by those nodes. The behavior of the override mechanism depends on the system configuration and security constraints.

The following nodes support a software override mechanism:

- RN-I
- RN-D
- CXHA
- MXP

RN-Is, RN-Ds, and CXHAs contain programmable registers to override the MPAM values for their own requests. The MXP contains programmable registers to override MPAM values for requests that are generated by RN-Fs.

The MXP software MPAM override mechanism is useful in the following situations:

- RN-F is a device that is compliant with CHI-D, and therefore supports MPAM, but MPAM is disabled in that device.
- RN-F is a device that does not support MPAM.

In these situations, the MXP MPAM override mechanism allows these devices to take part in the MPAM tracking scheme within the interconnect.

Each MXP contains an MPAM override register for each device port, por_mxp_p0_mpam_override, por_mxp_p1_mpam_override, por_mxp_p2_mpam_override, and por_mxp_p3_mpam_override. Software can configure these registers to generate an MPAM override value for request flits that are uploaded on that device port by RN-Fs. These registers are applicable only when an RN-F is connected to the port. For other device types, these registers are redundant and not used.

The following table shows the format of the override registers.

Table 4-15 por_mxp_p{0,1,2,3}_mpam_override register format

[63:25]	[24]	[23:17]	[16:8]	[7:5]	[4]	[3:1]	[0]
Reserved	REQ MPAM PMG	Reserved	REQ MPAM PartID	Reserved	REQ MPAM NS	Reserved	REQ MPAM override enable

The override enable bit indicates that the contents of the registers are valid, and therefore must be set when the other fields have the intended values.

The MXP MPAM override feature has the following security requirements:

- By default, the MPAM.NS subfield has the same value as the NS field of the REQ flit. When the override enable bit is set, the NS override field drives the NS field of the REQ flit.

The override behavior in the MXP depends on the CHI version that the RN-F supports and whether the enable bit is set.

When `CHI_MPAM_ENABLE` = 1, the RN-F CHI version is CHI-B or CHI-C, and the override enable bit is not set, then the override mechanism has the following behavior:

- All MPAM fields in the REQ flit, except the MPAM NS field, are driven to the default value, which is `0x00`.

When `CHI_MPAM_ENABLE` = 1, the RN-F CHI version is CHI-B or CHI-C, and the override enable bit is set, then all fields, including the MPAM NS field, are overridden.

When `CHI_MPAM_ENABLE` = 1, the RN-F CHI version is CHI-D, and the override enable bit is not set, then the request uses the original MPAM values of the REQ flit.

When `CHI_MPAM_ENABLE` = 1, the RN-F CHI version is CHI-D, and the override enable bit is set, then all MPAM fields are overridden with the contents of the register.

————— **Note** —————

When the RN-F CHI version is CHI-E, MPAM NS and Request NS do not need to match. For example, you can use Request NS = 0 and MPAM NS =1.

4.4.6 MTE support in HN-F

CMN-700 HN-F supports CHI-E *Memory Tagging Extensions* (MTE). For all coherent and non-coherent operations, if requests come with MTE opcodes, HN-F services them by fetching the required tags from SN-F.

In accordance with the CHI-E architecture, HN-F also flushes the dirty tags downstream to SN-F when required.

To disable the MTE handling in HN-F when the software is running without MTE opcodes, program the `hnf_mte_mode_dis` bit of the `por_hnf_cfg_ctl` register to 1. In this mode, HN-F ignores any requests for tags and drops dirty tags. For MTE Match requests, HN-F synthesizes a dummy Tag Match response to complete the protocol flow.

4.5 HN-F class-based resource allocation and arbitration

CMN-700 POCQ supports class-based resource allocation and arbitration for increased flexibility and control.

This section contains the following subsections:

- [4.5.1 Class assignment on page 4-1295](#).
- [4.5.2 POCQ resource allocation on page 4-1295](#).
- [4.5.3 POCQ request arbitration on page 4-1298](#).

4.5.1 Class assignment

QoS decoding takes place inside the HN-F.

A request from RN is assigned a class based on QoS or Request Type (Opcode) of a dynamic request. por_hnf_class_ctl config register provides configurability to determine this class assignment. Assigned class is then used for POCQ resource allocation and arbitration to SLC/SF access. It is also used when arbitrating for TXREQ.

QoS based class assignment

por_hnf_pocq_qos_class_ctl provides configurability to assign a class based on QoS min/max values.

————— Note —————

All QoS values must be assigned a class and assigned one and only one class.

Request Type (Opcode) based class assignment

Incoming request opcode can be classified as shown below:

Class 0

Read type requests

Class 1

Copy Back requests, including copyback wr+cmo

Class 2

Non Copy Back requests, including non-copyback wr+cmo

Class 3

All other requests

4.5.2 POCQ resource allocation

POCQ entries are allocated based on three configurable registers per class. These registers define different limits for each class.

por_hnf_pocq_alloc_class_dedicated

Dedicated entries are reserved entries for a given class in POCQ. For a given class, if number of requests in POCQ are less than dedicated, any incoming request to that class will not be retried. Unused dedicated entries belonging to a class are unavailable for other classes to use.

por_hnf_pocq_alloc_class_max_allowed

max_allowed indicates maximum number of entries allowed for any given class in POCQ. Once max_allowed is reached for a given class, any incoming request for that class will be retried, even if POCQ has available entries.

por_hnf_pocq_alloc_class_contented_min

contented_min indicates desired number of entries for a class under contention. Class is under contention when POCQ is full and request are retried. These entries are not reserved. Instead, requests for a class below its contented min are given higher priority than ones above contented min for credit grant arbitration.

————— Note ————

Total number of dedicated entries, including SEQ reserved, must not exceed number of POCQ entries.

For any given class, max allowed must not be less than dedicated entries.

For any given class, Arm recommends that:

- max_allowed > contended_min > dedicated

Retried requests are arbitrated per class to provide static credit grant back to requesting agent by the following priority order, highest to lowest:

- Request classes below dedicated with round robin arbitration across classes
- Request classes below contended min with round robin arbitration across classes
- Request classes below max_allowed in weighted round robin across classes

Weights for weighted round robin can be configured in “por_hnf_class_retry_weight_ctl”. Within a given class, static grants are allocated to different RNs based on round robin arbitration.

The following figures show possible POCQ resource allocation for various classes for dedicated and max_allowed. Dedicated is a minimum for a class and max_allowed is a maximum per class.

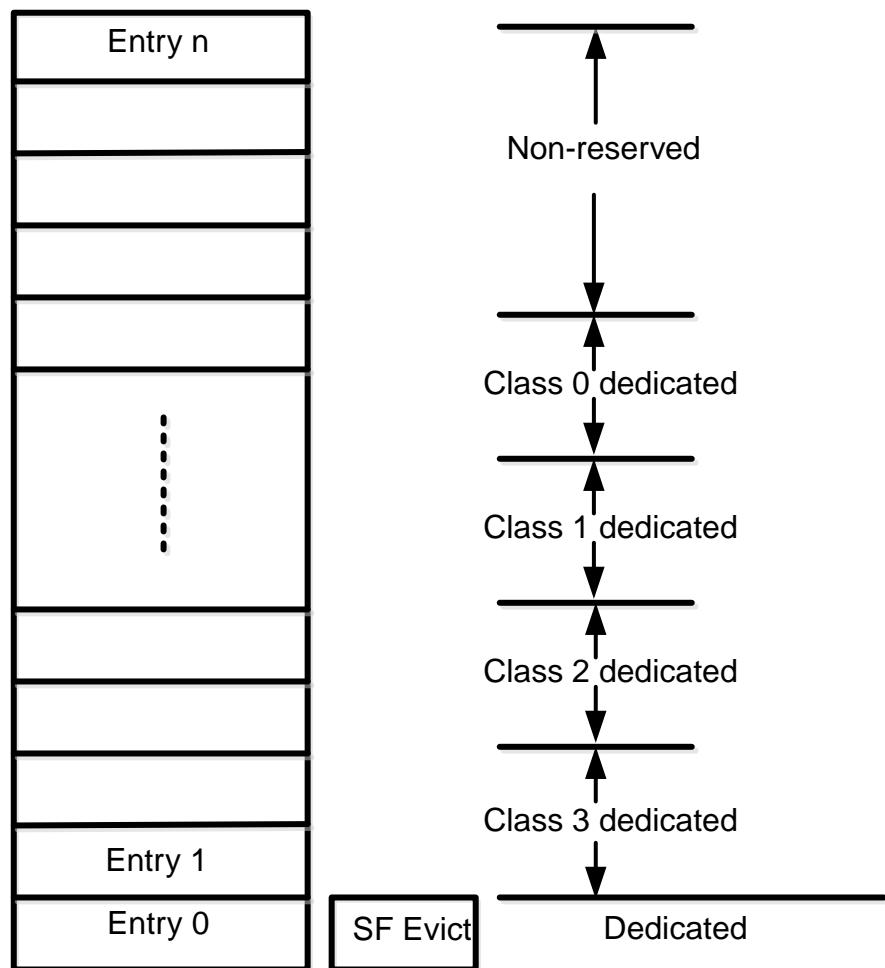


Figure 4-3 POCQ possible dedicated

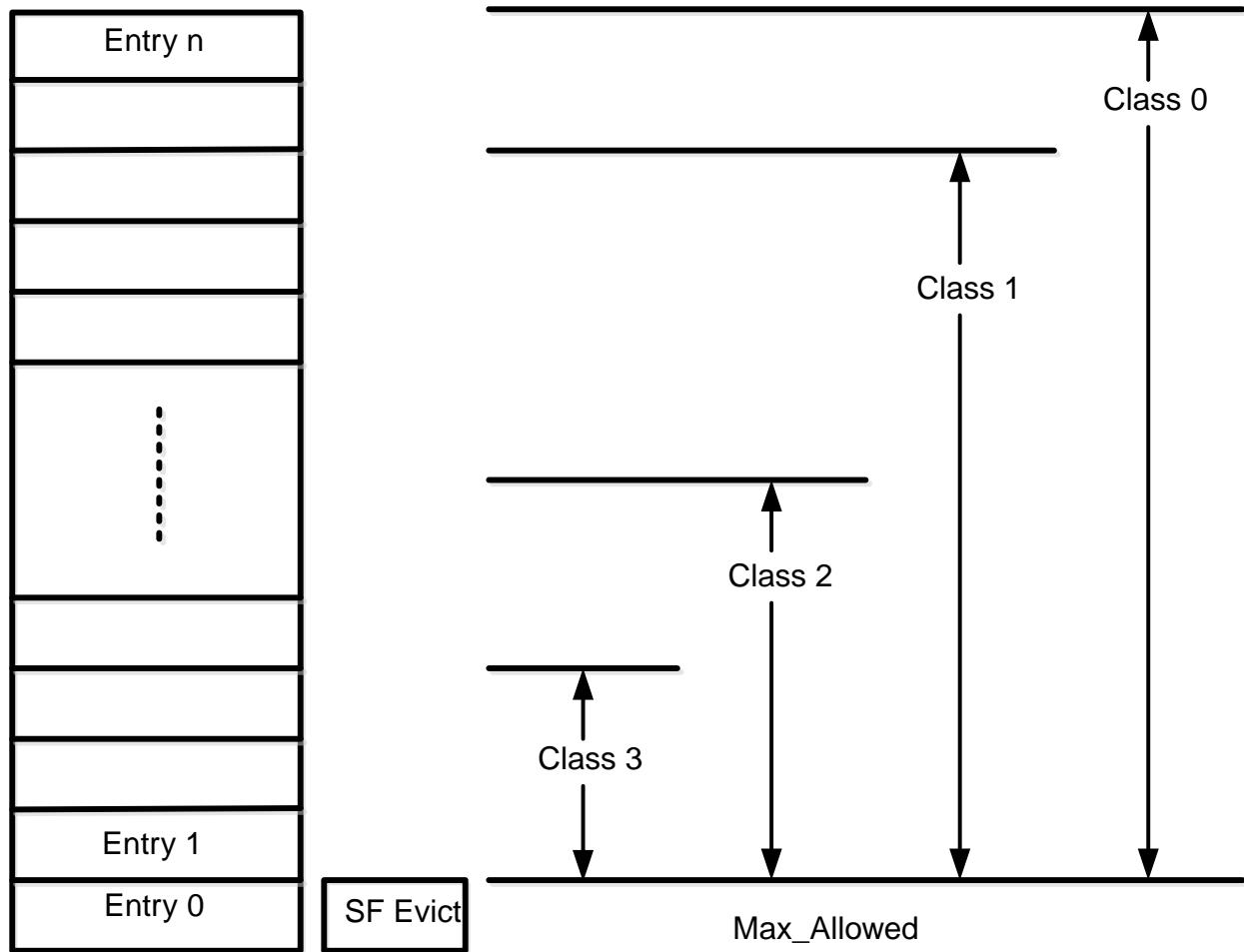


Figure 4-4 POCQ Default max_allocated

4.5.3 POCQ request arbitration

Requests from POCQ to SLC/SF pipe are selected by the following priority order, highest to lowest:

- Oldest requests in POCQ are ready for SLC/SF access
- Lookup requests: weighted round robin per class
- Any eligible requests: weighted round robin per class

Requests from POCQ to MC are selected by the following priority order, highest to lowest:

- Oldest requests in POCQ are ready to be sent to MC
- Static request: round robin per class
- All pending MC requests: weighted round robin per class

Configure the weights for weighted round robin in "por_hnf_class_pocq_arb_weight_ctl". The same weights are used for SLC/SF and MC weighted round robin.

Chapter 5

Debug Trace and PMU

This chapter describes the *Debug Trace* (DT) and *Performance Monitoring Unit* (PMU) features that CMN-700 implements.

It contains the following sections:

- [5.1 DT system overview](#) on page 5-1300.
- [5.2 DT usage examples](#) on page 5-1315.
- [5.3 PMU system overview](#) on page 5-1320.
- [5.4 CXLA PMU system](#) on page 5-1321.
- [5.5 Secure debug support](#) on page 5-1322.

5.1 DT system overview

CMN-700 provides at-speed self-hosted *Debug Trace* (DT) capabilities.

The CMN-700 DT capabilities include:

- Watchpoint-initiated and trace-tag-initiated transaction tracing.
- Globally synchronized cycle counters for precise tracing.
- CHI trace tag generation.
- CoreSight ATB trace streaming.
- Access to trace data through configuration registers.
- Cross trigger support.
- Secure debug support.
- Event-based interrupts.

The CMN-700 DT system consists of a set of *Debug Trace Controllers* (DTCs) and *Debug Trace Monitors* (DTMs) distributed across the interconnect. DTCs are located inside HN-Ds and HN-Ts while DTMs are located inside XPs.

All DTCs, including the master DTC, have an ATB interface and the following signals:

- **DBGWATCHTRIGREQ**.
- **DBGWATCHTRIGACK**.
- **INTREQPMU**.

The following signals are only present in the master DTC:

- **NIDEN**.
- **SPNIDEN**.
- **PMUSNAPSHOTREQ**.
- **PMUSNAPSHOTACK**.

————— Note ————

NIDEN and **SPNIDEN** are propagated from the master DTC to all DTMs. When asserted, they must remain asserted for at least 72 clock cycles. Likewise, when deasserted, they must remain deasserted for at least 72 clock cycles. This requirement ensures that all internal CMN-700 components transit into their debug and trace states correctly.

The following figure shows an example DT system with two DTC domains.

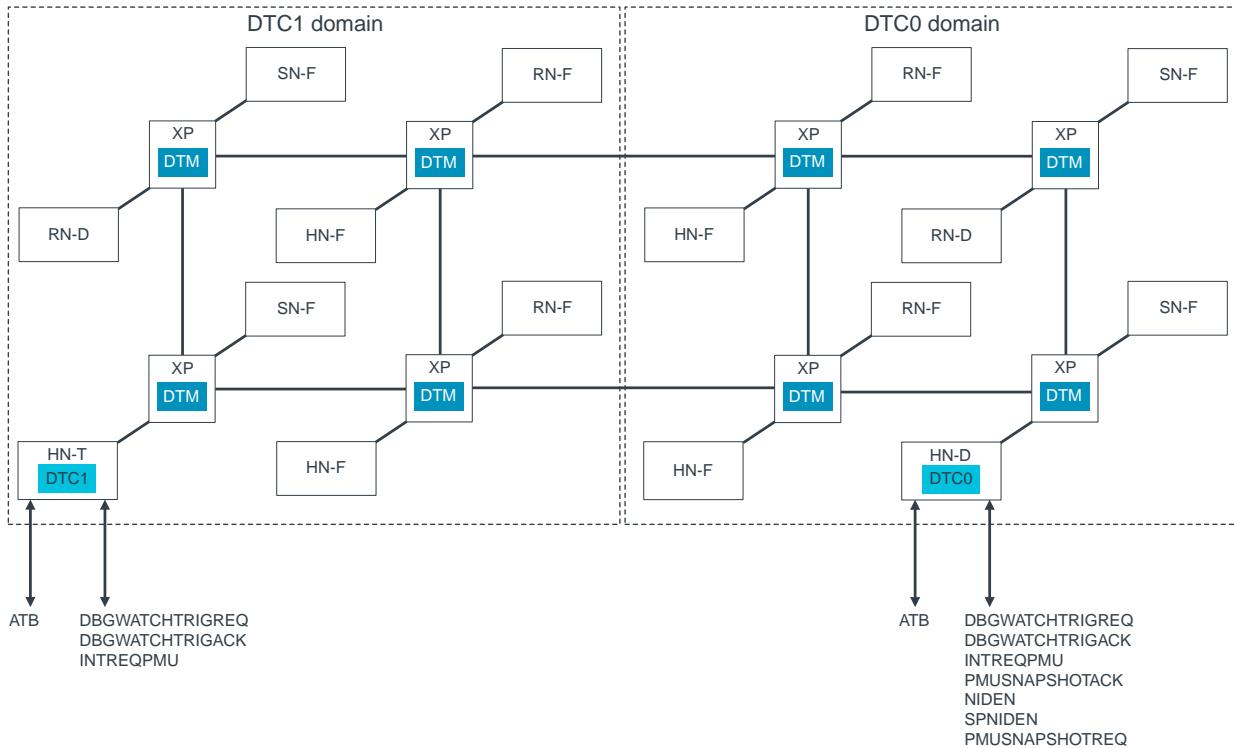


Figure 5-1 Example DT System with two DTC domains

————— Note ————

We recommend one DTC domain per 16 XPs.

————— Note ————

We recommend one DTC domain per 16 XPs. A combined maximum of 63 XP and CXRH DTM and PMUs are permitted in a single DTC domain. For a system with more than 63 XPs, at least one HN-T is required.

Each DTC is associated with a set of DTMs to form an exclusive DTC domain. When enabled, DTMs within a DTC domain collect trace data and transmit it to the associated DTC. The number of HN-T nodes in the mesh determines the number of DTC domains.

In a DT system comprising multiple DTCs, the DTC that is located inside the HN-D is designated as the master DTC or DTC0. You assign DTMs to DTCs by configuring XP parameters within Socrates IP Tooling platform.

Each DTC domain must be built using contiguous XPs.

Each DTC must have a contiguous XP for that domain.

The DT system implements the following functions:

- Monitoring of CHI flits at XP device ports using four sets of *WatchPoints* (WPs) in each DTM.
- Flit trace generation and storage at each DTM with control register access to trace packets.
- Trace tag generation.
- Debug trigger signaling and trace packet streaming over the *ATB* at each DTC.
- Internal event-based cross trigger generation and broadcast to all DTMs.
- Globally synchronized cycle counters.

————— Note ————

The DT system must not be active in mission critical mode.

This section contains the following subsections:

- [5.1.1 DTM watchpoint on page 5-1302](#).
- [5.1.2 DTM FIFO buffer on page 5-1306](#).
- [5.1.3 Read mode on page 5-1309](#).
- [5.1.4 DTC on page 5-1310](#).
- [5.1.5 ATB packets on page 5-1310](#).

5.1.1 DTM watchpoint

A DTM has four *WatchPoints* (WPs) that monitor flit uploads and downloads at XP device ports.

WPs monitor flits by matching on a subset of flit fields that you specify using a pair of val and mask registers. The following figure shows the WP comparator and the registers that control this functionality.

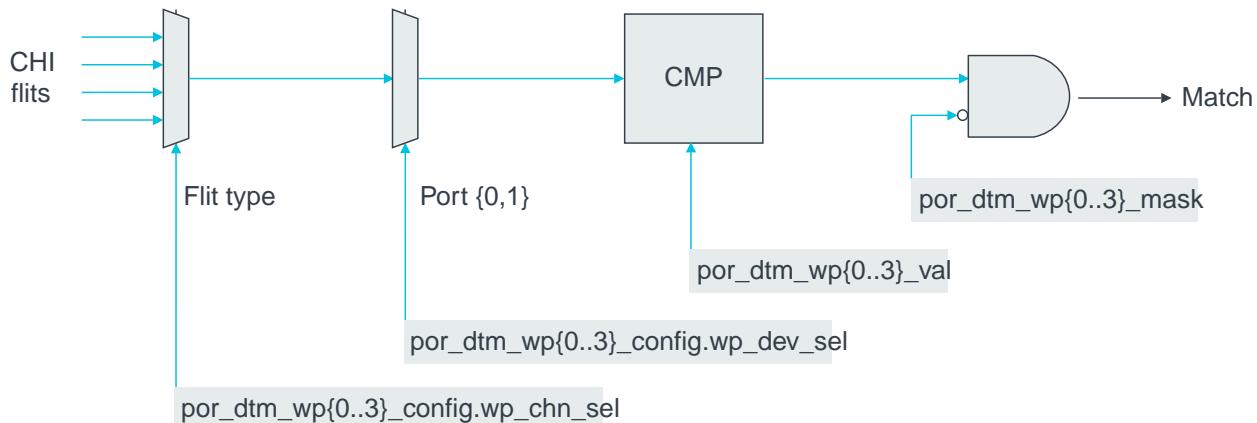


Figure 5-2 DTM WP comparator

A WP can be configured to monitor flits from one of two XP device ports and one of four CHI channels:

- REQ
- RSP
- SNP
- DAT

In addition, the WP can be configured to do one or more of the following tasks on detecting a flit match:

- Set trace tag bit on the flit.
- Generate flit trace.
- Generate cross trigger to DTC.
- Generate debug trigger to DTC.
- Increment PMU counters.

————— Note ————

Two WPs within a group can be combined for complex matching. For example, WP0 and WP1 can be combined, just as WP2 and WP3 can be combined.

The four DTM WPs are assigned to flit uploads and downloads according to the following groups:

- WP0 and WP1 are assigned to flit uploads.
- WP2 and WP3 are assigned to flit downloads.

See [Program DTM watchpoint on page 3-1253](#) for the DTM watchpoint programming sequence.

WP match value and mask register

The WP flit matching criteria are specified using a 64-bit match value register, por_dtm_wpN_val, and a 64-bit mask register, por_dtm_wpN_mask. These registers allow matching of up to 64 bits of the flit.

N = 0, 1, 2, or 3 for the match value and mask registers.

To specify the value for matching, write the value into the por_dtm_wpN_val register. The value of the por_dtm_wpN_mask register specifies the bits that must be masked from the match comparison, and therefore ignored. To specify that a bit must be masked, write a 1 into the corresponding bit position in the por_dtm_wpN_mask register.

The CHI flit fields are divided into the following match groups:

REQ channel

Primary, secondary, and tertiary match groups.

RSP channel

Primary match group.

SNP channel

Primary and secondary match group.

DAT channel

Primary and secondary match group.

The following tables specify the flit fields that belong to each of the groups for the different CHI channels.

Flit matching from two different match groups requires two WPs to be combined. For example, if both Opcode and Address fields of flits uploaded on the REQ channel are to be matched, then WP0 and WP1 must be combined, with Opcode match specified in WP0 and Address match specified in WP1 or the opposite way. Likewise, if both Opcode and Address fields of flits downloaded on the REQ channel must match, then WP2 and WP3 must be combined in a similar manner.

The following table shows REQ channel width and bit ranges for the primary match group.

Table 5-1 REQ channel: primary match group

Field	Width	Bit range
SRCID/TGTID	11	10:0
STASHNID/RETURNNID/Reserved[6:0]	11	21:11
StashTgtValid/Endian/Deep	1	22:22
{StashLPValid, StashLP[4:0]}	6	28:23
OPCODE	7	35:29
SIZE	3	38:36
NS	1	39:39
ALLOWRETRY	1	40:40
ORDER	2	42:41
PCRDTYPE	4	46:43
LPID	5	51:47
Reserved	3	54:52
EXPCOMPACK	1	55:55
RSVDC	8	63:56

The following table shows REQ channel width and bit ranges for the secondary match group.

Table 5-2 REQ channel: secondary match group

Field	Width	Bit range
QOS	4	3:0
ADDR	52	55:4
LIKELYSHARED	1	56:56
MEMATTR	4	60:57
SNPATTR	1	61:61
EXCL/SNOOPME	1	62:62
TRACETAG	8	63:63

The following table shows REQ channel width and bit ranges for the tertiary match group.

Table 5-3 REQ channel: tertiary match group

Field	Width	Bit range
SRCID/TGTID	11	10:0
OPCODE	7	17:11
MPAM	11	28:18
Reserved	2	30:29
ADDR[38:6]	33	63:31

The following table shows RSP channel width and bit ranges for the primary match group.

Table 5-4 RSP channel: primary match group

Field	Width	Bit range
QOS	4	3:0
SRCID/TGTID	11	14:4
OPCODE	5	19:15
RESPERR	2	21:20
RESP	3	24:22
FWDSTATE/DATAPULL	3	27:25
CBUSY	3	30:28
DBID	12	42:31
PCRDTYPE	4	46:43
Reserved	2	48:47
TRACETAG	1	49:49
DEVEVENT	2	51:50
Reserved	1	52:52
Reserved	1	53:53

The following table shows SNP channel width and bit ranges for the primary match group.

Table 5-5 SNP channel: primary match group

Field	Width	Bit range
SRCID	11	10:0
FWDTXNID/{2'b0, STASHLPIDVALID, STASHLPID[4:0]}/VMIDEXT[7:0]	8	18:11
FWDNID	11	29:19
OPCODE	5	34:30
NS	1	35:35
DONOTDATAPULL /DONOTGOTOSD	1	36:36
RETTOSRC	1	37:37
TRACETAG	1	38:38
QOS	4	42:39
MPAM	11	53:43

The following table shows SNP channel width and bit ranges for the secondary match group.

Table 5-6 SNP channel: secondary match group

Field	Width	Bit range
SRCID	11	10:0
ADDR	49	59:11

The following table shows DAT channel width and bit ranges for the primary match group.

Table 5-7 DAT channel: primary match group

Field	Width	Bit range
QOS	4	3:0
SRC/TGTID	11	14:4
HOMENID	11	25:15
OPCODE	4	29:26
RESPERR	2	31:30
RESP	3	34:32
DATAsrc/FWDSTATE/STASH	4	38:35
CBUSY	3	41:39
DBID	12	53:42
CCID	2	55:54
DATAID	2	57:56
POISON	4	61:58
DEVEVENT	2	63:62

The following table shows DAT channel width and bit ranges for the secondary match group.

Table 5-8 DAT channel: secondary match group

Field	Width	Bit range
SRC/TGTID	11	10:0
OPCODE	4	14:11
RESPERR	2	16:15
RESP	3	19:17
Reserved	2	21:20
Reserved	8	29:22
Reserved	2	31:30
DBID	12	43:32
TRACETAG	1	44:44
CHUNKV	2	46:45
DEVEVENT	2	48:47
RSVDC	8	56:49

5.1.2 DTM FIFO buffer

Traces captured by the DTM are stored in a four-entry DTM FIFO buffer. Each entry is 160-bits wide.

Entries are allocated to all enabled WPs as needed. Trace data from WPs are packed based on the trace data format and stored within each entry to efficiently use the limited buffer storage. Therefore an entry might contain trace data from multiple flits captured at different times.

As each FIFO entry is filled, trace data from that entry is sent to the DTC for streaming out through the ATB interface.

Trace data format

CMN-700 supports several trace data formats.

The 3-bit packet type encoding in the DTM WP configuration register (por_dtm_wp{0..3}_config.wp_pkt_type) specifies the trace data format. The following table provides the supported trace data formats and their packet type encodings.

Table 5-9 Trace data formats

Packet type	Trace data format	Size	Max traces per FIFO entry
000	TXNID[9:0]	10 bits	16
001	{OPCODE[5:0],TXNID[9:0]}	16 bits	10
010	{2'b00 {TGTID[10:0], SRCID[10:0], OPCODE[5:0], TXNID[9:0]}}	40 bits	4
011	Reserved	-	-
100	Control Flit (see the following tables for field descriptions)	REQ RSP SNP DAT	160 bits 68 bits 115 bits 94 bits
101	DATA[127:0]	-	-
110	DATA[255:128]	-	-
111	Reserved	-	-

Trace data is packed into a DTM FIFO buffer entry so that the higher-order bytes contain older trace data. For example, if the trace data format is set to TXNID (type 0) and three TXNIDs (trace data) are received in the order `0x01`, followed by `0x02`, followed by `0x03`, then the trace FIFO entry is set to:

- `0000_0000_0000_0000_0000_0000_0001_0203`

The following tables describe the control flit formats for various flit channels beginning with REQ control flit information, when MPAM is either enabled or disabled.

Table 5-10 REQ control flit

Field	Width	Bit range (MPAM enabled)	Bit range (MPAM disabled)
QOS	4	3:0	
TGTID	11	14:4	
SRCID	11	25:15	
TXNID	10	35:26	
STASHTGTID / RETURNNNID / Reserved[6:0]	11	46:36	
STASHTGTVALID /ENDIAN	1	47:47	
RETURNNTID[9:0] / {4'b0, STASHLPVALID, STASHLP[4:0]}	10	57:48	
OPCODE	6	63:58	
SIZE	3	66:64	
NS	1	67:67	
LIKELYSHARED	1	68:68	
ALLOWRETRY	1	69:69	
ORDER	2	71:70	
PCRDTYPE	4	75:72	
MEMATTR	4	79:76	
SNPATTR	1	80:80	
LPID	5	85:81	
EXCL/SNOOPME	1	86:86	
EXPCCOMPACK	1	87:87	
TRACETAG	1	88:88	
MPAM	11	99:89	-
ADDR	52	151:100	140:89
RSVDC	8	159:152	148:141
Total	160 (MPAM enabled) / 149 (MPAM disabled)	-	

The following table contains RSP control flit information.

Table 5-11 RSP control flit

Field	Width	Bit range
QOS	4	3:0
TGTID	11	14:4
SRCID	11	25:15
TXNID	10	35:26
OPCODE	4	39:36
FWDSTATE / DATAPULL	3	42:40
RESPERR	2	44:43
RESP	3	47:45
CBUSY	3	50:48
DBID	10	60:51
PCRDTYPE	4	64:61
TRACETAG	1	65:65
DEVEVENT	2	67:66
Reserved	1	68:68
Total	69	-

The following table contains SNP control flit information, when MPAM is either enabled or disabled.

Table 5-12 SNP control flit

Field	Width	Bit range (MPAM enabled)	Bit range (MPAM disabled)
QOS	4	3:0	
SRCID	11	14:4	
TXNID	10	24:15	
FWDNID	11	35:25	
FWDTXNID[9:0] / {4'b0, STASHLPVALID, STASHLP[4:0]} / {2'b00VMIDEXT[7:0]}	10	45:36	
OPCODE	5	50:46	
NS	1	51:51	
DONOTGOTOSD / DONOTDATAPULL	1	52:52	
RETTOSRC	1	53:53	
TRACETAG	1	54:54	
MPAM	11	65:55	-
ADDR	49	114:66	103:55
Total	115 (MPAM enabled) / 104 (MPAM disabled)	-	

See also [6.12 DEVEVENT](#) on page [6-1351](#) for more information.

The following table contains DAT control flit information.

Table 5-13 DAT control flit

Field	Width	Bit range
QOS	4	3:0
TGTID	11	14:4
SRCID	11	25:15
TXNID	10	35:26
HOMENID	11	46:36
OPCODE	4	50:47
RESPERR	2	52:51
RESP	3	55:53
FWDSTATE / DATAPULL	4	59:56
CBUSY	3	62:60
DBID	10	72:63
CCID	2	74:73
DATAID	2	76:75
TRACETAG	1	77:77
POISON	4	81:78
CHUNKV	2	83:82
DEVEVENT	2	85:84
RSVDC	8	93:86
Total	94	-

See [6.12 DEVEVENT on page 6-1351](#) for more information.

————— **Note** —————

CHUNKV[1:0] denotes whether the upper or lower 128 bits of data are valid.

5.1.3 Read mode

Read mode provides an alternate way to access trace data that is stored in the DTM trace FIFO buffer, through configuration register access.

Each entry in the FIFO buffer is mapped to three 64-bit configuration registers, dtm_fifo_entry{0..3}_X, where X = 0, 1, or 2.

Read mode is enabled by setting the trace_no_atb bit in the DTM control register (por_dtm_control). Setting this bit causes all FIFO entries to be cleared and the DTM FIFO read status register (por_dtm_fifo_entry_ready) to be reset.

In this mode, each FIFO entry is allocated to the corresponding WP. For example, por_dtm_fifo_entry0_{0..2} is allocated to WP0 and por_dtm_fifo_entry1_{0..2} is allocated to WP1.

For packet types 0b000, 0b001, and 0b010, only TXNID, OPCODE, SRCID, and TGTID are accumulated in the FIFO. The oldest transactions are on the MSB side of the FIFO,

por_dtm_fifo_entry<0..3>_2. The newest transactions are on LSB side of the FIFO,
por_dtm_fifo_entry<0..3>_0.

————— Note —————

When you program any DTM in read mode, the ATB protocols, for example flush, do not function properly.

The read status for each WP trace data is reflected in the corresponding bit in the DTM FIFO entry ready status register (por_dtm_fifo_entry_ready). When a FIFO entry is full, the corresponding status bit is set, indicating that the trace data is ready to be read. Subsequent writes into that FIFO entry are disabled until the status bit is cleared. A write of 1 clears the status bit and enables the corresponding FIFO entry to capture subsequent trace data.

5.1.4 DTC

DTCs control DTMs.

The main features of the DTC are:

- Trace packing, generation, and streaming through ATB interface.
- Time stamping of traces.
- Global synchronized cycle counters in all units (16-bit).
- ATB flush of DTM and DTC.
- ATB flush of DTM and DTC (**AFREADY** might be asserted several cycles after trace data output as DTC must receive all flush responses from DTMs).
- Watchpoint trigger event-based interrupt.
- Eight sets of performance counters (32-bit) with shadow registers, which are paired with one or more DTM local counters.
- PMU snapshot of DTM and DTC.
- PMU overflow interrupt.

see [Program DTC on page 3-1254](#) for the DTC programming sequence.

5.1.5 ATB packets

Each DTC has an ATB interface and generates ATB packets to send downstream through this interface. There are different varieties of ATB packets which are used for different functions.

Each DTC aggregates flit trace data from the DTMs into the DTC trace FIFO, packetizes them, and sends them out on its ATB interface. The DTCs also send other control and debug packets through this interface. There are various packet formats that are used on the ATB interface, which are described in the following sections:

- [Trace data packet format on page 5-1310](#)
- [Alignment sync packet format on page 5-1311](#)
- [Time stamp packet format on page 5-1312](#)
- [Cycle counting packet format on page 5-1312](#)
- [Trace stream example on page 5-1313](#)

Trace data packet format

Trace data packet contains a 4B header and a payload of variable size.

The following figure shows the packet header.

VC	DEV	WP#	Type				Byte 3
Size			node ID[10:8]				
node ID[7:0]							
0	1					CC	lossy

Figure 5-3 Trace data packet header

The packet header contains the following fields:

VC CHI channel.

00	REQ
01	RSP
10	SNP
11	DAT

DEV Device port number (0 or 1).

WP# Watchpoint number that captured the trace (0-3).

Type Packet format type.

Size Payload size, which is specified as (number of bytes – 1).

NodeID CHI node ID, shifted right by 3 bits reflecting the (X,Y) coordinates of the XP where the trace was captured.

CC Cycle counter. When set, this field indicates that a 2B cycle count is included in the packet after the payload.

The following key points must be observed:

- For packet type 100, the leading zeros in upper-order bytes of the trace data payload are compressed and not transmitted. The payload Size field in the trace packet header is adjusted accordingly. For example, trace data = 0000_0000_0000_0000_0000_0000_0001_0203 is sent as 01_0203, with Size = 2 (indicating 3B transferred).
- For packet type 100, 101, and 110 the leading zeros in upper-order bytes of the trace data payload are compressed and not transmitted. The payload Size field in the trace packet header is adjusted accordingly. For example, trace data = 0000_0000_0000_0000_0000_0000_0001_0203 is sent as 01_0203, with Size = 2 (indicating 3B transferred).
- The WP field selection for match might be different from the type of payload that is generated from the matching. When WPs are combined, the lower watchpoint number is specified as the WP# in the trace packet header.
- Trace data is of variable length. The expected number of bytes, not including the header, is (Size + 1). And with CC, another 2B are included at the end of the trace data.
- Any time the previous packets cannot be transmitted in full, a lossy bit is asserted for the immediate next packet. A separate lossy bit is maintained for each of the watchpoints.

Alignment sync packet format

The alignment sync delimits trace start.

The alignment sync packet is 16B long and comprises 9B of zeros followed by 0x80.

The alignment sync packet is 20B long and comprises 19B of zeros followed by 0x80.

The alignment sync packet is 20B long and comprises 15B of zeros followed by 0x80.

The alignment sync packet is the first packet that is sent after tracing is enabled. Also, you can configure the DTC to send the alignment sync packet periodically by programming the por_dt_trace_control register.

Time stamp packet format

The time stamp packet carries the SoC timer information. The time stamp is used to align the sequence of trace events across the SoC.

The time stamp packet is sent opportunistically under the following circumstances when the DTC FIFO has enough space to accommodate the time stamp packet:

- After each alignment sync packet is sent.
- When flush is complete.
- Periodically based on the setting of the timestamp_period field of the por_dt_trace_control register and only when trace packets have been sent after the last time stamp packet.

The following figure shows the time stamp packet format.



Figure 5-4 Time stamp packet

The time stamp packet contains the following fields:

TS# 3-bit encoding of the size of time stamp that is specified as (number of bytes – 1).

CC When set, indicates that a 2B cycle count is included in the packet after the payload.

Cycle counting packet format

Trace packets include an optional attached cycle counter.

Each watchpoint includes a configuration bit. The logical operator AND is used on the configuration bit and global cycle count enable. The following figure shows a typical cycle counting scenario.

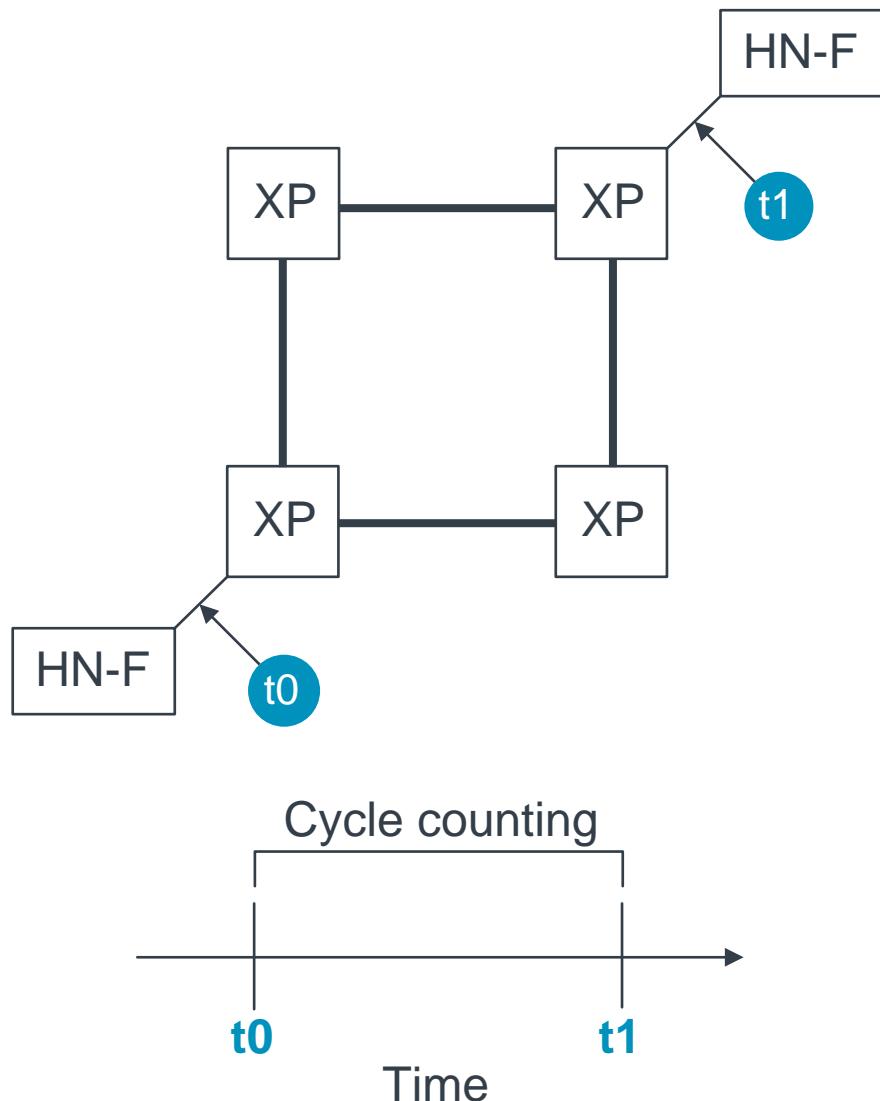


Figure 5-5 Cycle counting

The cycle counter payload is 2B and is indicated by the CC bit in the trace packet header. Cycle counters across the interconnect are turned on and off synchronously. This feature ensures that all of them have the same time stamp value.

Trace stream example

DTCs send out trace data on the ATB bus as a trace stream.

The following figure shows an example trace stream. It consists of:

- 4B trace packet header.
- $<M>B$ trace data.
- 2B cycle count.
- 1B time stamp header.
- $<N>B$ time stamp.
- 2B cycle count.

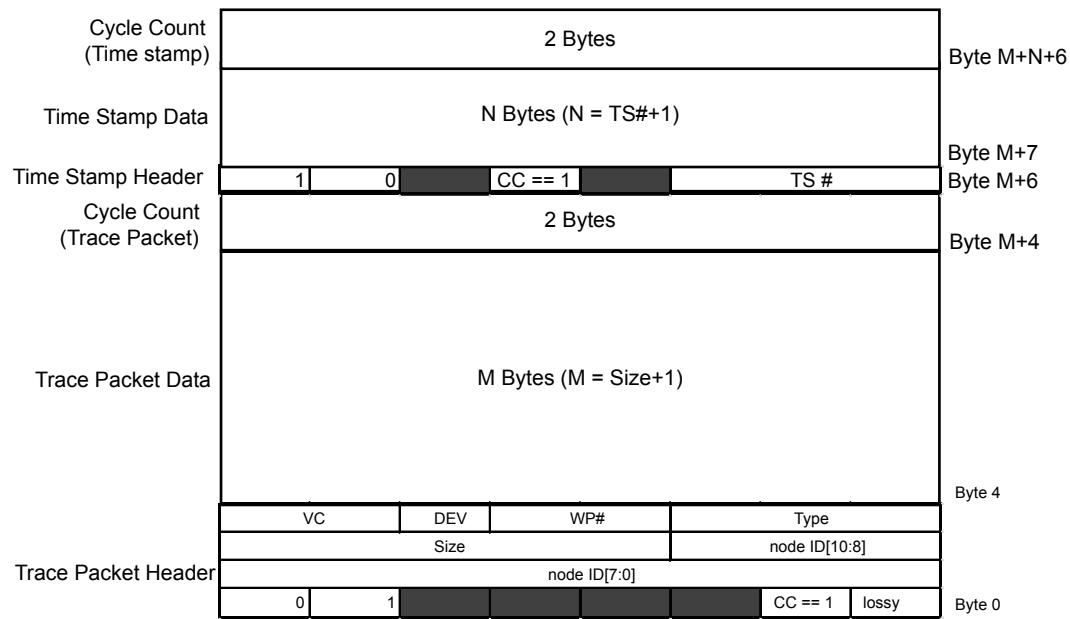


Figure 5-6 Trace stream

5.2 DT usage examples

To help you use the CMN-700 DT features, we describe some example use cases of the DT system and example programming for those use cases.

For more information, see the following sections:

- [5.2.1 Flit tracing on page 5-1315](#)
- [5.2.2 Trace tag on page 5-1316](#)
- [5.2.3 Debug watch trigger events on page 5-1318](#)
- [5.2.4 Cross trigger on page 5-1318](#)

This section contains the following subsections:

- [5.2.1 Flit tracing on page 5-1315](#).
- [5.2.2 Trace tag on page 5-1316](#).
- [5.2.3 Debug watch trigger events on page 5-1318](#).
- [5.2.4 Cross trigger on page 5-1318](#).

5.2.1 Flit tracing

CMN-700 can trace individual flits at device interfaces at each XP.

You can program DTM WPs to monitor flit uploads and downloads at each of the two XP device ports on any of the four CHI channels:

- REQ
- RSP
- SNP
- DAT

You can use a set of value and mask registers to define a subset of flit fields that are then used for matching at the WP.

On a match, WPs capture and store flit fields into trace buffers so that they can be used for debug. The generated trace can then be streamed out on the ATB interface or accessed using a control register interface.

For more information about the format of the value and mask registers, and the format of trace packets, see [WP match value and mask register on page 5-1303](#) and [Trace data format on page 5-1306](#).

Flit tracing example

CMN-700 can trace individual flits at device interfaces at each XP.

For more information, refer to [Program DTM watchpoint on page 3-1253](#).

This section shows an example of setting up a simple trace of REQ flits. The example corresponds to a ReadShared transaction to address=X initiated by RNF2 and sending out the trace packets on the ATB bus.

To monitor REQ flits uploaded from RNF2, set up watchpoints (WPs) inside XP connected to RNF2. The Opcode and Address fields are mapped to the primary and secondary match registers respectively. Therefore, you must set up two WPs, one to monitor the Opcode and the other to monitor the Address.

To set up these WPs:

1. Program WP0 (upload WP) to monitor REQ.Opcode:
 - a. Set dtm_wp0_val/mask registers to match on Opcode=ReadShared
 - b. Set dtm_wp0_config to:
 - a. Select upload device port (wp_dev_sel=RNF2_port).
 - b. Select upload device port ({wp_dev_sel2, wp_dev_sel}=RNF2_port).
 - c. Select flit channel (wp_chn_sel=REQ)
 - d. Match format group to primary for Opcode match (wp_grp=0)

- e. Set combined mode to gang-up WP0 and WP1 (wp_combine=1)
- f. Enable REQ flit trace packet generation (set wp_pkt_type and wp_pkt_gen=1)
2. Program WP1 (upload WP) to monitor REQ.Address as follows:
 - a. Set dtm_wp1_val/mask registers to match on Address=X
 - b. Set dtm_wp1_config to:
 - a. Select upload device port (wp_dev_sel=RNF2_port).
 - b. Select upload device port ({wp_dev_sel2, wp_dev_sel}=RNF2_port).
 - c. Select Flit channel (wp_chn_sel=REQ)
 - d. Match format group to secondary for Address match (wp_grp=1)

Combined mode and trace generation

Note

In combined mode, WP0 config settings are used to enable trace generation.

To enable trace generation in the WP, program WP0 as follows:

1. Set dtm_control.dtm_enable = 1 and program the DTC to start the trace
2. Program por_dt_traceid.traceid according to the *Arm® CoreSight™ Architecture Specification*. The supported range of values is 0x01-0x6F
3. Program por_dt_dtc_ctl to enable tracing (dt_en = 1)

5.2.2 Trace tag

CMN-700 can generate trace tags at the device interfaces and propagate them to destination devices.

This feature enables a set of flits corresponding to a specific transaction or set of transactions that match a specific criterion to be tagged for tracing.

For example, using the trace tag mechanism, flits from all four CHI channels can be monitored:

- REQ
- RSP
- SNP
- DAT

An example of a monitored transaction is a memory read transaction to a specific address, which is then tagged for tracing.

Trace tag generation

A trace tag is generated either internally by an XP or externally by an RN-F or SN-F device. The generated trace tag is reflected in the TRACETAG field of the uploaded flit.

Inside the XP, DTM WPs generate the trace tag by matching on flits uploaded at the corresponding XP device port. The DTM WPs can be programmed to match on a flit on any of the four CHI channels: REQ, RSP, DAT, and SNP.

The WP generates a trace tag when there is a match and the trace_tag_enable field of the por_dtm_control register is set to 1.

You can program DTM WPs to match on a flit on any CHI channel and on any device port. However, for debug, it is most useful to program WPs to match on REQ flits uploaded at the RN and HN-F device ports. We recommend this programming because REQ flits are the starting flits that originate new transactions. When tagged, subsequent RSP, SNP, and DAT flits that relate to the same transaction carry the trace tag.

If the following conditions are all true, the XP does not generate the trace tag:

- Flit transfer takes place from one device port to the other device port, within the same XP.
- The flit destination device is not an HN.
- The flit transfer to its destination occurs one cycle after the XP receives it.

Trace tag propagation

All CMN-700 devices forward on the trace tag, when asserted, from a received flit corresponding to a transaction to all subsequent flits associated with that transaction.

The HN-F also propagates the trace tag from the source transaction to spawned transactions such as SLC evictions and SF back invalidations.

Using the logical operator OR, the trace tag that is generated inside the XP is combined with the TRACETAG field of the received flit. The resultant value is then sent in the TRACETAG field of the flit transmitted to the destination device.

Trace tag trace packet generation

When a watchpoint is enabled for trace packet generation through wp_pkt_gen, there are several interactions. Any flit with the TRACETAG field asserted on the channel (which wp_chn_sel selects), for the device (which wp_dev_sel has selected), generates a trace packet type which wp_pkt_type selects.

This trace packet is generated whenever TRACETAG is asserted in a flit; a watchpoint match, determined by wp_val and wp_mask, is not required.

Trace tag example programming

This example programming outlines a trace tag scenario-based trace generation with synchronized cycle counts.

For more information about watchpoint programming, refer to [Program DTM watchpoint on page 3-1253](#).

This example programming sets up a simple trace of REQ flits. The example corresponds to a ReadShared transaction to address=X. RN-F 2 initiates the transaction in the mesh and the WP sends trace packets out on the ATB bus.

To monitor REQ flits uploaded from RN-F 2, set up WPs inside the XP that RN-F 2 is connected to. For REQ flits, the Opcode and Address fields are mapped to the primary and secondary match registers respectively. Therefore, you must set up two WPs, one to monitor the Opcode and the other to monitor the Address.

To set up these WPs:

1. Program WP0 (upload WP) to monitor REQ.Opcode:
 - a. Set por_dtm_wp0_val and por_dtm_wp0_mask registers to match on Opcode=ReadShared.
 - b. Set por_dtm_wp0_config to:
 - a. Select upload device port (wp_dev_sel=RN-F 2 port).
 - b. Select upload device port ($\{wp_dev_sel2, wp_dev_sel\}$ =RN-F 2 port).
 - c. Select flit channel (wp_chn_sel=REQ).
 - d. Match format group to primary for Opcode match (wp_grp=0).
 - e. Set combined mode to gang-up WP0 and WP1 (wp_combine=1).
 - f. Enable REQ flit trace packet generation (set wp_pkt_type and wp_pkt_gen=1).
2. Program WP1 (upload WP) to monitor REQ.Address:
 - a. Set por_dtm_wp1_val and por_dtm_wp1_mask registers to match on Address=X.
 - b. Set por_dtm_wp1_config to:
 - a. Select upload device port (wp_dev_sel=RN-F 2 port).
 - b. Select upload device port ($\{wp_dev_sel2, wp_dev_sel\}$ =RN-F 2 port).
 - c. Select flit channel (wp_chn_sel=REQ).
 - d. Match format group to secondary for Address match (wp_grp=1).

In combined mode, the configuration settings for WP0 are used to enable trace generation, using the following steps:

1. Enable trace tag generation in the WP:

- Set the trace_tag_enable field of the por_dtm_control register = 1.
 - Set the dtm_enable field of the por_dtm_control register = 1 and program the DTC to start the trace.
2. Program the traceid field of the por_dt_traceid register according to the *Arm® CoreSight™ Architecture Specification*. The supported range of values is 0x01-0x6F.
 3. Program por_dt_dtc_ctl to enable tracing (dt_en = 1).

5.2.3 Debug watch trigger events

DTM WPs can be programmed to match on specific flits and generate a debug watch trigger event to the DTC.

You can program the DTC to signal the debug watch trigger event in one or both of the following ways:

- Signal a debug watch trigger interrupt on the **DBGWATCHTRIGREQ/DBGWATCHTRIGACK** interface.

————— Note ————

This interface is based on a four-phase handshake protocol.

- Signal an ATB trace trigger with ATID 0x7D on the ATB interface.

In a system with multiple DTCs, each DTC has its own ATB interface on which it signals ATB trace triggers from DTM within its DTC domain. Multiple DTCs also have their own **DBGWATCHTRIGREQ / DBGWATCHTRIGACK** interfaces on which they signal debug watch trace interrupts.

5.2.4 Cross trigger

CMN-700 can trigger DTMs based on specific events occurring elsewhere in the system.

By default, DTMs start monitoring and tracing flits without waiting for another event. The cross trigger feature allows flit monitoring and tracing to be delayed until after the events of interest are observed in the system.

The cross trigger event is set up in two steps:

1. Set up DTM WPs to monitor flits and generate traces.
2. Other DTM WPs (in the same XP or different XPs) are set up to generate a cross trigger on specific events to the DTC which is programmed to trigger the DTMs in step 1.

Cross trigger example

CMN-700 can trigger DTMs based on specific events occurring elsewhere in the system.

For more information, refer to [Program DTM watchpoint on page 3-1253](#).

This example uses trace DAT flits corresponding to a ReadShared transaction to address-X that originated at RN-F2. There have also been 10 WriteNoSnoops uploaded to the HN-D.

1. Set WP or WPs at all DAT download ports to generate DAT flit traces for ReadShare transactions from RN-F2 to address-X.
 - a. Program WP2 and WP3, which are at the DAT download ports, to trace DAT flits:
 - a. Set dtm_wp2_val/mask and dtm_wp3_val/mask registers to match on SRCID = RN-F2, opcode = ReadShared, and address = X.
 - b. Set dtm_wp2_config and dtm_wp3_config to the respective download device ports (wp_dev_sel = 0 and wp_dev_sel = 1). Select the DAT channel by setting the wp_chn_sel bit to the DAT encoding.
 - b. Enable the WP by setting the dtm_enable bit of the dtm_control register to 1.
2. Set up WP at HN-D upload port to monitor WriteNoSnoop flits.
 - a. Program WP0 (upload WP) to monitor and enable cross trigger REQ. Opcode as follows:
 - a. Set dtm_wp0_val/mask registers to match on Opcode = WriteNoSnoop.
 - b. Set dtm_wp0_config to:

- a. Select upload device port (wp_dev_sel = HND_port).
 - b. Flit channel (wp_chn_sel = REQ).
 - c. Match format group to primary for Opcode match (wp_grp = 0).
 - d. Enable cross trigger (wp_ctrig_en = 1).
- b. Enable WP.
 - Set dtm_control.dtm_enable = 1.
3. Set up counter in DTC to count ten trigger events from step 3.
 - Program por_dt_dtc_ctl as follows:
 1. Set cross trigger count (cross_trigger_count = 10).
 2. Enable waiting for HN-D WP trigger event (dt_wait_for_trigger = 1).
 3. Enable DTC (dt_en = 1).

Sample profile

CMN-700 supports the Armv8.2 sample extension.

WPs can be programmed to monitor channel, opcode, and related PM items. A Sample Interval Counter Register (PMSICR) counts down with each match. When the counter reaches zero, the trace tag of the next matched transaction is asserted. At the same time, the counter is reloaded with the programmed value from the PMSIRR register, and the next count down cycle begins.

There is only one set of PMSCIR and PMSIRR registers per XP, as only one outstanding transaction is expected. PMSCIR is 24 bits, and the lower 8 bits of PMSIRR are zero.

In general, Secure transactions are allowed to be tagged and traced with the secure_debug_disable field of the por_dt_secure_access register. When this bit is set, Secure registers are read with Non-secure access.

5.3 PMU system overview

CMN-700 includes *Performance Monitoring Unit* (PMU) capabilities.

The PMU offers the following features:

- Local and global performance counters with shadow registers.
- PMU snapshot across all internal CMN-700 devices.

The PMU consists of local performance counters in the DTM^s and global performance counters in the DTC. The following figure shows this structure of local and global performance counters.

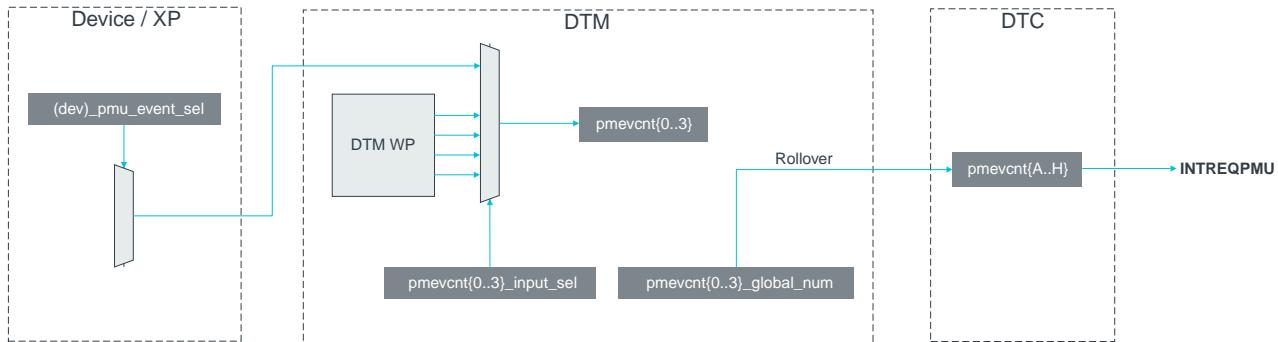


Figure 5-7 PMU local and global performance counters

For the various PMU programming sequences, see [3.4.9 PMU system programming](#) on page 3-1254.

The PMU system performs the following tasks:

- Selects PMU event from XP, the local watchpoint, and the devices on XP device ports.
- Operates four local PMU counters ($4 \times 16b$).
- Operates eight global PMU counters ($8 \times 32b$) associated with the local counters.
- Snapshot.
- Overflow interrupt from global PMU counters.

The PMU counter value can be copied over into the shadow registers when there is:

- A request of snapshot through input pin **PMUSAPSHOTREQ**.
- A write into the *ss_req* field of the [por_dt_pmsrr](#) on page 3-865 register within the DTC.

On receiving a snapshot request, a DTC sends a snapshot request to all DTMs. Multiple snapshot requests are collapsed into a single request. On receiving PMU snapshot packets from all DTMs, rollover information is updated at the global counters, and the counter value is copied to the shadow registers.

5.4 CXLA PMU system

The CXLA PMU system is present across multiple nodes in the CXG. There are certain details that you must be aware of for the system to function correctly.

The CXLA PMU system has two components: the PMU controller, `por_cxg_pmu_ctl`, and the PMU counter, `por_cxla_pmu`.

For the CXLA PMU system to function correctly, the power status of CXLA must be communicated to both DTC and the local CXLA PMU controller. Set the `por_cxg_ra_cfg_ctl.en_cxla_pmucmd_prop` register field to 1 if CXLA is in the powered up state, and CXLA PMU is functioning. The number of CXLAs that are powered up is added to the number of DTMs to specify the total number of PMU counters.

————— **Note** —————

The total number of XP and CXRH DTMs and PMUs in a single DTC domain must not be greater than 63.

————— **Note** —————

The PMU events in CXLA are captured in the `por_cxla_pmevcnt` registers and HA/RA PMU events are captured in XP the `por_dtm_pmevcnt` register, in the same way as all other devices.

5.5 Secure debug support

The **SPNIDEN** input and the value of the `secure_debug_disable` field of the `por_dt_secure_access` register control the Secure debug state.

Secure debug is enabled when **SPNIDEN** is asserted, or when the `secure_debug_disable` bit of the `por_dt_secure_access` register is set to 0. The default value of this bit is 0.

When Secure debug is enabled, all events can be counted and all flits can be traced.

When Secure debug is disabled, all events with UNKNOWN Secure state are not counted and all flits with UNKNOWN Secure state are not traced.

Chapter 6

Performance optimization and monitoring

This chapter describes the performance optimization techniques and *Performance Monitoring Unit* (PMU) that System integrators can use to optimize the functionality of the interconnect implementation.

It contains the following sections:

- [6.1 Performance optimization guidelines on page 6-1324](#).
- [6.2 About the Performance Monitoring Unit on page 6-1326](#).
- [6.3 HN-F performance events on page 6-1327](#).
- [6.4 RN-I performance events on page 6-1333](#).
- [6.5 SBSX performance events on page 6-1337](#).
- [6.6 MTSX performance events on page 6-1341](#).
- [6.7 HN-I performance events on page 6-1343](#).
- [6.8 DN performance events on page 6-1347](#).
- [6.9 XP PMU event summary on page 6-1348](#).
- [6.10 CXG performance events on page 6-1349](#).
- [6.11 Occupancy and lifetime measurement using PMU events on page 6-1350](#).
- [6.12 DEVEVENT on page 6-1351](#).

6.1 Performance optimization guidelines

There are some restrictions when optimizing CMN-700.

To obtain maximum performance from CMN-700, the system integrator must be aware of the following information:

RN-I

When request ordering is not required, transaction requests must be dispatched with non-overlapping IDs to ensure optimal bandwidth operation. Large burst transactions, that is, larger than 64B, must be split into 64B or smaller burst transactions. In addition, set **AxSIZE** to the AXI bus width of the RN-I to fully utilize the available bandwidth.

For example, if the AXI bus width is:

- 128b** Set **AxSIZE** = 4 (16B).
- 256b** Set **AxSIZE** = 5 (32B).
- 512b** Set **AxSIZE** = 6 (64B).

Read or Write requests to different parts of the same cache line must be combined into a single cache line request. For example, multiple (partial) WriteUnique transactions must be combined into a single WriteUnique or a single WriteLineUnique transaction, where all bytes in the cache line are written.

Based on the transaction attributes, RN-I can enforce more ordering on transactions, targeting device memory downstream of HN-I, affecting the overall achievable bandwidth.

RN-I and RN-D also support disabling of read data interleaving using the `s*_dis_data_interleaving` configuration bits in the `por_rnd_s_0-2_port_control` registers. This setting only applies to data returned on the **RDATA** channel in response to read requests from AR channels. Disabling of read data interleaving does not apply to atomic operations. For example, the **RDATA** data from an atomic operation sent on the AW channel can interleave with **RDATA** data from an AR channel request.

RN-I and RN-D upsizes a sub-cacheline ReadOnce transaction arriving on AXI to a 64B ReadOnce request on CHI which may result in extra data beats on CMN Interconnect. To prevent data overhead, RN-I and RN-D added a feature to change opcode to ReadNoSnoop on CHI while keeping original transaction size. Note that this feature relies on Invisible cache behavior of HNF. By default, this feature is not enabled if target type is CXRA. If it is known that all CXRA's in the system are connected to SMP CCIX link, then a bit in RNI cfg ctrl register (`en_subcacheline_rdonce_conv_to_cxra`) must be programmed to enable this feature for CXRA.

RN-I and RN-D support preservation of AXI burst for PCIe read transactions through the Interconnect when the request is targeting HNP and PCI_CXRA node types (see [6.1.2 PCIe Read Burst Preservation through Interconnect on page 6-1325](#)).

RN-I and RN-D also support AXID-based targetID selection for HN-P, CXRA, PCI_CXRA targets (see [6.1.3 RN-I/RN-D AXID Based target selection on page 6-1325](#)).

HN-F

High temporal locality of address usage in transactions can cause same-address dependencies to occur for transactions with addresses to overlapping cache lines. This condition results in higher latency because of serialization delays between these transactions. CMN-700 is microarchitected to avoid hot spotting in the HN-F partitions or in the memory controllers, but this condition is unavoidable in cases of temporally local same-address usage.

HN-I

Stream of interleaved reads and writes targeting the same peripheral downstream of HN-I results in higher latencies on these transactions. It also could result in serialization delays between these reads and

writes. Arm recommends that read and write transactions are not interleaved when targeting the same peripheral.

This section contains the following subsections:

- [6.1.1 RN-I and RN-D write burst cracking on page 6-1325](#).
- [6.1.2 PCIe Read Burst Preservation through Interconnect on page 6-1325](#).
- [6.1.3 RN-I/RN-D AXID Based target selection on page 6-1325](#).

6.1.1 RN-I and RN-D write burst cracking

RN-I and RN-D crack write bursts into individual CHI transactions.

In HN-P, **AWUSER[MSB]** is used to indicate the last transaction of a burst from RN-I or RN-D. The last indication and **AWID** can be used to gather write transactions from a burst downstream of HN-P.

6.1.2 PCIe Read Burst Preservation through Interconnect

RN-I/RN-D preserves AXI burst for PCIe read transactions through the interconnect when the request is targeting HNP and PCI_CXRA node types under following conditions :

- RNID_NUM_RD_REQ == RNID_NUM-RD-BUF
- RNID_AXDATA_WIDTH is 512 or 256
- ARCACHE[1] == 1'b1
- ARBURST == INCR
- ARSIZE == RNID_AXDATA_WIDTH
- Read transactions must be Uniq-ID which is detected using:
 - ARIDUNQ indication on Transaction
 - Setting config bit in RNID per port:
- s#{index}_sends_ar_inq_id to 1 (x=0/1/2)
- RN-I/RN-D Config Control register bit, PCIE_MASTER_PRESENT, must be set to 1
- RN-I/RN-D Port Control register bit, s#{index}_dis_data_interleaving, must be set to 1

————— Note —————

1. Read Burst preservation feature can be disabled by setting the config bit, DIS_RD_BURST, in RNI/RND, to 1. When DIS_RD_BURST=1, AXI bursts are cracked into 64B chunks.
2. When a request doesn't target HNP and PCI_CXRA or doesn't follow the restrictions mentioned in 5.1.2, bursts are cracked into 64B chunks.

6.1.3 RN-I/RN-D AXID Based target selection

The RN-I/RN-D RN SAM can be configured to enable AXID-based targetID selection for HN-P, CXRA, PCI_CXRA targets.

This enables higher bandwidth by striping traffic across multiple targets. TargetID within Hashed Target Group (HTG) in RNSAM can be programmed in different order for different RN-I/RN-D for a given address range. This avoids targeting same TargetID when two or more RN-I/RN-D receive traffic with overlapping AXIDs. See RN SAM section for more details

6.2 About the Performance Monitoring Unit

CMN-700 provides access to various performance events. Some of these events are unique to and originate in a specific CMN-700 component, and some are available by using watchpoints in the DTM watchpoint in the XP where the component is located.

The PMU input source must be configured to select the watchpoint input, according to the instructions in [Set up PMU counters on page 3-1254](#).

This chapter describes the performance events and the relevant use cases for most of those events. For information about the infrastructure and logic that enable general utility of the performance monitor events, see [Chapter 5 Debug Trace and PMU on page 5-1299](#).

The following table shows the PMU events.

6.2.1 Cycle counter

The cycle counter is used to track time.

You can reset this counter to initiate the time interval over which you want to capture the events.

PMU_CYCLE_COUNTER Cycle counter.

GCLK0 clocks the cycle counter. Therefore, the cycle counter is not incremented during periods of HCG when the clocks are stopped.

The global clock signal or signals clock the cycle counter. Therefore, the cycle counter is not incremented during periods of HCG when the clocks are stopped.

6.3 HN-F performance events

The HN-F performance analysis counters are used to monitor cache behavior.

For a particular cache, the cache miss or hit rate is used to measure the capacity of the cache, and the location for certain applications. To measure the cache miss rate, the performance monitor counters count the number of instances of cache accesses and cache misses.

This section contains the following subsections:

- [6.3.1 Cache performance on page 6-1327](#).
- [6.3.2 HN-F counters on page 6-1328](#).
- [6.3.3 SF events on page 6-1328](#).
- [6.3.4 System-wide events on page 6-1329](#).
- [6.3.5 Snoop events related to SF clustering on page 6-1329](#).
- [6.3.6 Quality of Service on page 6-1330](#).
- [6.3.7 HN-F PMU event summary on page 6-1330](#).

6.3.1 Cache performance

Cache performance events are required to calculate the cache miss rate and the cache allocation.

HN-Fs support MPAM-related PMU events. See the *Arm® Architecture Reference Manual Supplement Memory System Resource Partitioning and Monitoring (MPAM)*, for Armv8-A for more information about configuration.

The following sections describe the cache performance events.

Cache miss rate

The cache events that are required to calculate the cache miss rate are:

PMU_HN_CACHE_MISS_EVENT

Counts the total cache misses. A miss results from a first-time lookup and is high priority.

PMU_HNSLC_SF_CACHE_ACCESS_EVENT

The total number of cache accesses. An access is first-time and high priority.

Note

The performance counter architecture allows up to four HNs to collect the cache miss rate for each DTC domain. In a system with multiple DTC domains, more than four HNs can collect the cache miss rate. However, because of the CMN-700 microarchitecture, the cache miss rate that is measured at one HN-F within an SCG is a good proxy for the cache miss rate of the remaining HN-Fs.

Note

The performance counter architecture enables only four HNs to collect the cache miss rate. However, because of the CMN-700 microarchitecture, the cache miss rate that is measured at one HN-F is a good proxy for the cache miss rate of the remaining HN-Fs.

Calculate the cache miss rate as follows:

$$\text{Cache miss rate (\%)} = \frac{\text{Total cache misses}}{\text{Total cache accesses}} \times 100$$

Figure 6-1 Cache miss rate

Certain request types can cause multiple cache accesses:

- Lookup.
- Tag update.

- Victim selection.
- Cache fill.

Event counting is therefore limited to first time accesses only. For example, for a ReadUnique transaction that leads to an SLC hit, PMU_HNSLC_SF_CACHE_ACCESS_EVENT is only counted the first-time cache lookup is performed. The tag update is not counted as a cache access. Similarly, for WriteBack or Write*Unique transactions with an SLC allocate hint, only the first instance of an SLC lookup is counted as an access and hit or miss. The eventual victim selection and cache fill are not counted as further accesses.

Cache allocations

The cache allocation event counts the number of times an HN-F SLC cache is allocated. It provides an approximate cache usage for this particular application over a specific time slice. This event does not check whether the application has any hot sets.

PMU_HN_CACHE_FILL_EVENT Counts all cache line allocations to SLC cache.

All cache line writes, that is, Write*Unique, WriteBack, and Evictions that are allocated in SLC cache, are counted towards this event.

6.3.2 HN-F counters

Applications can bottleneck on one or more HN-Fs because they frequently target an address or a stream of addresses.

The following POCQ occupancy and request retry events are used to monitor possible performance loss in the system:

PMU_HN_POCQ_RETRY_EVENT The total number of requests that have been retried.

PMU_HN_POCQ_REQS_RECV_EVENT The total number of requests that the HN-F receives.

Requests that cannot be queued in the POCQ, because of lack of credits, are retried. The HN-F responds with a RetryAck response, and the request waits for a static credit. This wait period indicates whether a lack of credits is causing the bottlenecks, and also shows if the latency of requests is very high.

Calculate the message retry rate as follows:

$$\text{HN-F message retry rate (\%)} = \frac{\text{HN-F total messages retried}}{\text{HN-F total messages received}} \times 100$$

Figure 6-2 HN-F message retry rate

6.3.3 SF events

There are three snoop events that can be counted.

The following sections describe the SF performance events.

SF miss rate

This event measures the amount of memory controller traffic that is generated. It can also be used to measure the efficiency of the SF.

PMU_HN_SF_HIT_EVENT Measures the number of SF hits.

Calculate the SF hit rate as follows:

$$\text{Snoop filter hit rate (\%)} = \frac{\text{Total snoop filter hits}}{\text{Total SLC lookups}} \times 100$$

Figure 6-3 SF hit rate

SF accesses are only counted for first-time lookups, and not for the victim selection accesses or SF fills. Because the SLC lookup and SF lookups are parallel, the SLC lookups can be used to calculate the SF hit rate.

SF evictions

This event measures the frequency of SF evictions.

PMU_HN_SF_EVICTIONS_EVENT	Measures the number of SF evictions when cache invalidations are initiated.
----------------------------------	---

Snoops sent and received with hit rate

These events measure the amount of shared data across clusters for a specific application, using snoop hits or misses.

PMU_HN_SNOOPS_SENT_EVENT	Number of snoops sent. Does not differentiate between broadcast or directed snoops.
PMU_HN_SNOOPS_BROADCAST_EVENT	Number of snoop broadcasts sent.

Calculate the snoops sent and received rate as follows:

$$\text{Shared data (\%)} = \frac{\text{Total snoops broadcast}}{\text{Total snoops sent}} \times 100$$

Figure 6-4 Sent and received snoops rate

The number of broadcast and total snoops measures the shared data invalidations.

6.3.4 System-wide events

The memory controller request retries determine whether the memory controller is the bottleneck in the system, which can cause higher request latencies.

The following events can be counted:

PMU_HN_MC_RETRIES_EVENT	Number of requests that are retried to the memory controller.
PMU_HN_MC_REQS_EVENT	Total number of requests that are sent to the memory controller.

Calculate the retry rate for requests to the memory controller as follows:

$$\text{MC message retry rate (\%)} = \frac{\text{MC total messages retried}}{\text{MC total messages received}} \times 100$$

Figure 6-5 MC message retry rate

6.3.5 Snoop events related to SF clustering

Certain HN-F PMU events can be used to understand the performance impact of SF clustering.

The following events can be counted:

PMU_HN_SNP_SENT_CLUSTER_EVENT	Counts the number of snoops that are sent at the level of a whole cluster. This event does not count individual snoops within a cluster. For example, in a cluster with four RN-Fs, if HN-F sends four snoops, this event counts these four snoops as one since all four snoops are sent to the same cluster.
PMU_HN_SF_IMPRECISE_EVICT_EVENT	Counts the number of times an Evict operation from an RN does not clear the SF tracking because the line was in shared state (imprecise). If there is a single RN-F in the cluster, then the Evict operation always clears the tracking for that RN-F. However, in SF clusters, the Evict operation must not clear the SF tracking as other RN-Fs might still be accessing this line.
PMU_HN_SF_EVICT_SHARED_LINE_EVENT	Counts the number of times an SF eviction happened to a cache line that was in shared state. This event can be helpful in understanding of the impact of SF pollution in clustered mode.

6.3.6 Quality of Service

Requests with a HighHigh QoS must be allocated and processed from the POCQ with the highest priority compared to High, Medium, and Low QoS requests.

If the HighHigh requests are retried too frequently, there could be a bottleneck at a particular HN-F, or the POCQ reservation for HighHigh requests requires adjustment.

PMU_HN_QOS_HH_RETRY How often a HighHigh request is retried.

6.3.7 HN-F PMU event summary

The following table shows a summary of the HN-F PMU events.

Table 6-1 HN-F events

Number	Name	Description
1	PMU_HN_CACHE_MISS_EVENT	Counts total cache misses in first lookup result (high priority).
2	PMU_HNSLC_SF_CACHE_ACCESS_EVENT	Counts number of cache accesses in first access (high priority).
3	PMU_HN_CACHE_FILL_EVENT	Counts total allocations in HN SLC (all cache line allocations to SLC).
4	PMU_HN_POCQ_RETRY_EVENT	Counts number of retried requests.
5	PMU_HN_POCQ_REQS_RECVED_EVENT	Counts number of requests that HN receives.
6	PMU_HN_SF_HIT_EVENT	Counts number of SF hits.
7	PMU_HN_SF_EVICTIONS_EVENT	Counts number of SF eviction cache invalidations initiated.
8	PMU_HN_DIR_SNOOPS_SENT_EVENT	Counts number of directed snoops sent (not including SF back invalidation).
9	PMU_HN_BRD_SNOOPS_SENTEVENT	Counts number of multicast snoops sent (not including SF back invalidation).
10	PMU_HN_SLC_EVICTION_EVENT	Counts number of SLC evictions (dirty only).

Table 6-1 HN-F events (continued)

Number	Name	Description
11	PMU_HN_SLC_FILL_INVALID_WAY_EVENT	Counts number of SLC fills to an invalid way.
12	PMU_HN_MC_RETRIES_EVENT	Counts number of retried transactions by the MC.
13	PMU_HN_MC_REQS_EVENT	Counts number of requests that are sent to MC.
14	PMU_HN_QOS_HH_RETRY_EVENT	Counts number of times a HighHigh priority request is protocol-retried at the HN-F.
15	PMU_HNF_POCQ_OCCUPANCY_EVENT	Counts the POCQ occupancy in HN-F. Occupancy filtering is programmed in pmu_occup1_id.
16	PMU_HN_POCQ_ADDRHAZ_EVENT	Counts number of POCQ address hazards on allocation.
17	PMU_HN_POCQ_ATOMICS_ADDRHAZ_EVENT	Counts number of POCQ address hazards on allocation for atomic operations.
18	PMU_HN_LD_ST_SWP_ADQ_FULL_EVENT	Counts number of times ADQ is full for Ld/St/SWP type atomic operations while POCQ has pending operations.
19	PMU_HN_CMP_ADQ_FULL_EVENT	Counts number of times ADQ is full for CMP type atomic operations while POCQ has pending operations.
20	PMU_HN_TXDAT_STALL_EVENT	Counts number of times HN-F has a pending TXDAT flit but no credits to upload.
21	PMU_HN_TXRSP_STALL_EVENT	Counts number of times HN-F has a pending TXRSP flit but no credits to upload.
22	PMU_HN_SEQ_FULL_EVENT	Counts number of times requests are replayed in SLC pipe due to SEQ being full.
23	PMU_HN_SEQ_HIT_EVENT	Counts number of times a request in SLC hit a pending SF eviction in SEQ.
24	PMU_HN_SNP_SENT_EVENT	Counts number of snoops sent including directed, multicast, and SF back invalidation.
25	PMU_HN_SFBI_DIR_SNP_SENT_EVENT	Counts number of times directed snoops were sent due to SF back invalidation.
26	PMU_HN_SFBI_BRD_SNP_SENT_EVENT	Counts number of times multicast snoops were sent due to SF back invalidation.
27	PMU_HN_SNP_SENT_UNTRK_EVENT	Counts number of times snoops were sent due to untracked RNFs.
28	PMU_HN_INTV_DIRTY_EVENT	Counts number of times SF back invalidation resulted in dirty line intervention from the RN.
29	PMU_HN_STASH_SNP_SENT_EVENT	Counts number of times stash snoops were sent.
30	PMU_HN_STASH_DATA_PULL_EVENT	Counts number of times stash snoops resulted in data pull from the RN.
31	PMU_HN_SNP_FWDED_EVENT	Counts number of times data forward snoops were sent.
32	PMU_HN_SNP_SENT_CLUSTER_EVENT	Counts number of snoops sent to clusters excluding individual snoops within a cluster.
33	PMU_HN_SF_IMPRECISE_EVICT_EVENT	Counts number of times an evict operation was dropped due to SF clustering.

Table 6-1 HN-F events (continued)

Number	Name	Description
34	PMU_HN_SF_EVICT_SHARED_LINE_EVENT	Counts number of times a shared line was evicted from SF.
35	PMU_HN_POCQ_CLASS_OCCUPANCY_EVENT	Counts the given POCQ occupancy for a given class in HN-F; Class occupancy filtering is programmed in pmu_class_occup_id
36	PMU_HN_POCQ_CLASS_RETRY_EVENT	Counts number of retried requests for a given class; Class filtering is programmed in pmu_class_occup_id
37	PMU_HN_CLASS_MC_REQS_EVENT	Counts number of requests sent to MC for a given class; Class filtering is programmed in pmu_class_occup_id
38	PMU_HN_CLASS_PCRDGNT_BELOW_CONDMIN_EVENT	Counts number of protocol credit grants for a given class when it's above dedicated and below conditional min; Class filtering is programmed in pmu_class_occup_id
39	PMU_HN_NUM_SN_CBUSY_THROTTLE_EVENT	Counts number of times request to SN was throttled due to cbusy; Event filtering is programmed in pmu_cbusy_snthrottle_sel
40	PMU_HN_NUM_SN_CBUSY_THROTTLE_MIN_EVENT	Counts number of times request to SN was throttled to the minimum due to cbusy; Event filtering is programmed in pmu_cbusy_snthrottle_sel

6.4 RN-I performance events

External devices connect at an RN-I bridge.

This section contains the following subsections:

- [6.4.1 Bandwidth at RN-I bridges on page 6-1333](#).
- [6.4.2 Bottleneck analysis at RN-I bridges on page 6-1334](#).
- [6.4.3 RN-I PMU event summary on page 6-1335](#).

6.4.1 Bandwidth at RN-I bridges

External devices connect at an RN-I bridge.

The following events measure bandwidth at the RN-I bridges:

- [Requested read bandwidth at RN-I bridges on page 6-1333](#).
- [Actual read bandwidth on interconnect on page 6-1333](#).
- [Write bandwidth at RN-I bridges on page 6-1334](#).

Requested read bandwidth at RN-I bridges

External devices connect to CMN-700 at an RN-I bridge.

To monitor the behavior of the system, the following events measure the read bandwidth at each RN-I bridge:

RDataBeats_Port0 Number of RData beats, **RVALID** and **RREADY**, dispatched on port 0.

RDataBeats_Port1 Number of RData beats, **RVALID** and **RREADY**, dispatched on port 1.

RDataBeats_Port2 Number of RData beats, **RVALID** and **RREADY**, dispatched on port 2.

Because CMOs are sent through the read channel, their responses are included in these events.

Calculate the read bandwidth as follows:

$$\text{Read bandwidth} = \frac{\text{Number RDataBeats_Portn} \times \text{AXIDataBeatSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-6 Read bandwidth calculation

Where AXIDataBeatSize is the number of bytes for each AXI beat. Usually, this number is the same size as the AXI bus.

————— Note ————

If the data chunking feature is enabled, Read data bandwidth is calculated by counting the number of chunks being transferred on RData. This count is done by looking at **RCHUNKSTRB[n-1:0]** signal, where every bit of **RCHUNKSTRB** represents 16B of data.

Actual read bandwidth on interconnect

RXDATFLITV measures the bandwidth that an RN-I bridge sends to the interconnect.

This event counts the number of received data flit requests that the bridge receives through the data channel. Therefore, this event measure the actual bandwidth that an RN-I bridge sends to the interconnect, and not the useful bandwidth the external devices can use.

RXDATFLITV Number of **RXDAT** flits received. This event is a measure of the true read data bandwidth. It excludes CMOs, because CMO completions return to the RN-I through the response channel, but includes replayed requests.

This event includes the replayed requests because of the read data buffer decoupled scheme.

Calculate the actual read bandwidth as follows:

$$\text{Actual read bandwidth} = \frac{\text{RXDATFLITV} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-7 Actual read bandwidth

Write bandwidth at RN-I bridges

TXDATFLITV monitors the number of data flits that the RN-I bridge sends out.

In a similar way to the read actual bandwidth event, this event monitors the number of data flits that the RN-I bridge sends out. Therefore, this event measures the actual write bandwidth that is sent to the interconnect:

TXDATFLITV	Number of TXDAT flits dispatched. This event is a measure of the write bandwidth.
-------------------	--

Calculate the write bandwidth as follows:

$$\text{Actual write bandwidth} = \frac{\text{TXDATFLITV} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-8 Actual write bandwidth

6.4.2 Bottleneck analysis at RN-I bridges

CMN-700 provides events that observe the locations where the nodes or bridges are full, which can cause delays in the rest of the system.

This feature enables you to monitor the current bottlenecks in the system, and checks multiple events in the RN-Is, HN-Fs, and memory controllers. In the RN-I bridges, the events monitor the following:

- The number of times the bridge is forced to retry because of the lack of dynamic credits.
- The number of times the read and write tracker is full and therefore cannot accept new requests in the system. This condition can cause delays in the AXI masters.
- The number of read request replays, because of decoupling of the read request buffers and read data buffers in the RN-I system.

Request retry rate at RN-I bridges

TXREQFLITV_RETRYED monitors the efficiency of using dynamic credits in the system.

It does this task by measuring the request retry rate:

TXREQFLITV_RETRYED	Number of retried TXREQ flits dispatched. This event is a measure of the retry rate.
---------------------------	---

Calculate the request retry rate as follows:

$$\text{Retry rate} = \frac{\text{TXREQFLITV}_\text{RETRYED}}{\text{TXREQFLITV}_\text{TOTAL}}$$

Figure 6-9 Retry rate

Read and write delays at RN-I bridges

To monitor the delays for both reads and writes, CMN-700 enables you to monitor how full the read and write trackers are in the RN-I bridges.

When one of the trackers is full, the bridge cannot accept new requests from the AXI master. This condition delays the I/O devices that connect to the AXI master.

There are two measures that, together, can help you to isolate the source of bottlenecks in the system. These measures are: how full the trackers are, and the read and write bandwidth from the RN-I bridge to the interconnect. For example:

- If the read tracker of a specific RN-I bridge is full but the effective read bandwidth from the bridge is not close to the maximum expected, the interconnect cannot keep up with the read traffic from the specific device.
- If the bandwidth is close to maximum, the I/O device can send requests to the maximum of its port bandwidth and the tracker is full for this reason.

You can also use the measure of how full the trackers are with AXI PMUs to monitor delays to the AXI masters.

The following events monitor the read and write trackers:

RRT_OCCUPANCY All entries in the read request tracker are occupied. This event is a measure of oversubscription in the read request tracker.

WRT_OCCUPANCY All entries in the write request tracker are occupied. This event is a measure of oversubscription in the write request tracker.

Note

For CMN-700 when the NUM_RD_REQ parameter for an RN-I or RN-D node is configured to 128 or 256, the read tracker is divided into slices of 64 entries each. An ACE-Lite request is allocated into a particular read tracker slice based on a hash of the ARID value of the request and which of the three ACE-Lite slave interfaces receives the request. Therefore, in these configurations, the maximum number of outstanding same-ARID requests from the same ACE-Lite slave interface is 64.

The RRT_OCCUPANCY event covers the total occupancy of all read tracker slices.

6.4.3 RN-I PMU event summary

There are 16 RN-I PMU events.

The following table shows a summary of the RN-I PMU events.

Table 6-2 RN-I PMU event summary

Number	Name	Description
1	PMU_RNI_RDATABASEATS_P0	Number of RData beats, RVALID and RREADY , dispatched on port 0. This event measures the read bandwidth, including CMO responses.
2	PMU_RNI_RDATABASEATS_P1	Number of RData beats, RVALID and RREADY , dispatched on port 1. This event measures the read bandwidth, including CMO responses.
3	PMU_RNI_RDATABASEATS_P2	Number of RData beats, RVALID and RREADY , dispatched on port 2. This event measures the read bandwidth, including CMO responses.
4	PMU_RNI_RXDATFLITV	Number of RXDAT flits received. This event measures the true read data bandwidth, excluding CMOs.
5	PMU_RNI_TXDATFLITV	Number of TXDAT flits dispatched. This event measures the write bandwidth.
6	PMU_RNI_TXREQFLITV	Number of TXREQ flits dispatched. This event measures the total request bandwidth.
7	PMU_RNI_TXREQFLITV_RETRY	Number of retried TXREQ flits dispatched. This event measures the retry rate.
8	PMU_RNI_RRT_OCCUPANCY	All entries in the read request tracker are occupied. This event measures oversubscription in the read request tracker.
9	PMU_RNI_WRT_OCCUPANCY	All entries in the write request tracker are occupied. This event measures oversubscription in the write request tracker.
10	PMU_RNI_TXREQFLITV_REPLAYED	Number of replayed TXREQ flits. This event measures the replay rate.
11	PMU_RNI_WRCANCEL_SENT	Number of write data cancels sent. This event measures the write cancel rate.

Table 6-2 RN-I PMU event summary (continued)

Number	Name	Description
12	PMU_RNI_WDATABASEAT_P0	Number of WData beats, WVALID and WREADY , dispatched on port 0. This event measures write bandwidth on AXI port 0.
13	PMU_RNI_WDATABASEAT_P1	Number of WData beats, WVALID and WREADY , dispatched on port 1. This event measures the write bandwidth on AXI port 1.
14	PMU_RNI_WDATABASEAT_P2	Number of WData beats, WVALID and WREADY , dispatched on port 2. This event measures the write bandwidth on AXI port 2.
15	PMU_RNI_RRTALLOC	Number of allocations in the read request tracker. This event measures the read transaction count.
16	PMU_RNI_WRTALLOC	Number of allocations in the write request tracker. This event measures the write transaction count.
17	PMU_RNI_RDB_UNORD	Number of cycles for which Read Data Buffer state machine is in Unordered Mode.
18	PMU_RNI_RDB_REPLAY	Number of cycles for which Read Data Buffer state machine is in Replay mode.
19	PMU_RNI_RDB_HYBRID	Number of cycles for which Read Data Buffer state machine is in hybrid mode. Hybrid mode is where there is a mix of ordered and unordered traffic.
20	PMU_RNI_RDB_ORD	Number of cycles for which Read Data Buffer state machine is in ordered Mode.

6.5 SBSX performance events

This section contains SBSX performance event information.

This section contains the following subsections:

- [6.5.1 Bandwidth at SBSX bridges on page 6-1337](#).
- [6.5.2 Bottleneck analysis at SBSX bridges on page 6-1338](#).
- [6.5.3 SBSX PMU event summary on page 6-1340](#).

6.5.1 Bandwidth at SBSX bridges

This section contains SBSX bridge bandwidth information.

The following events are used to measure bandwidth at the SBSX bridges:

- [Read bandwidth on interconnect at SBSX bridges on page 6-1337](#).
- [Write bandwidth at SBSX bridges on page 6-1337](#).
- [Total requested bandwidth at SBSX bridges on page 6-1338](#).

Read bandwidth on interconnect at SBSX bridges

This section contains information on read bandwidth on interconnect at SBSX bridges.

This event counts the number of received data flits at the SBSX and interconnect:

PMU_SBSX_RXDAT Number of RXDAT flits received at XP from SBSX. This event is a measure of the read data bandwidth.

Calculate the actual read bandwidth as follows:

$$\text{Actual read bandwidth} = \frac{\text{PMU_SBSX_RXDAT} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-10 Actual read bandwidth

————— Note ————

This event is tracked in the DTM watchpoint in the XP where the component is located.

————— Note ————

This event is tracked in the DWM, not in the SBSX, and is defined from the XP's perspective.

Write bandwidth at SBSX bridges

This section contains information on write bandwidth at SBSX bridges.

In a similar way to the read actual bandwidth event, this event monitors the number of data flits that the SBSX receives. Therefore, this event measures the actual write bandwidth that is received from the interconnect:

PMU_SBSX_TXDAT Number of TXDAT flits dispatched from XP to SBSX. This event is a measure of the write bandwidth.

Calculate the write bandwidth as follows:

$$\text{Actual write bandwidth} = \frac{\text{PMU_SBSX_TXDAT} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-11 Actual write bandwidth

————— **Note** ————

This event is tracked in the DTM watchpoint in the XP where the component is located.

————— **Note** ————

This event is tracked in the DWM, not in the SBSX design. The event is defined from the perspective of the XP.

Total requested bandwidth at SBSX bridges

This section contains information on total requested bandwidth at SBSX bridges.

To improve efficiency when using PMU events and signals, this event combines the read and write bandwidth estimation in a single event. The PMU_SBSX_TXREQ_TOTAL event monitors the number of REQ flits that an SBSX bridge receives:

PMU_SBSX_TXREQ_TOTAL

Number of TXREQ flits dispatched from XP to SBSX. This event is a measure of the total request bandwidth.

Calculate the total bandwidth as follows:

$$\text{Total requested bandwidth} = \frac{\text{PMU_SBSX_TXREQ_TOTAL} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-12 Total requested bandwidth

————— **Note** ————

This event is tracked in the DTM watchpoint in the XP where the component is located.

————— **Note** ————

This event is tracked in the DWM, not in the SBSX design, and is defined from the perspective of the XP.

6.5.2 Bottleneck analysis at SBSX bridges

This section contains information on bottleneck analysis at SBSX bridges.

CMN-700 provides events that observe the locations where the nodes or bridges are full, which can cause delays in the rest of the system. This feature enables you to monitor the current bottlenecks in the system, and checks multiple events in all CMN-700 components. The events monitor the following:

- The number of times the bridge is forced to retry because of the lack of dynamic credits.
- The number of cycles the bridge is forced to stall due to backpressures on AXI or CHI interface.

The following events are used to measure bottlenecks at the SBSX bridges:

- [6.5 SBSX performance events on page 6-1337](#).

Request retry rate at SBSX bridges

This section contains information on the request retry rate at SBSX bridges.

RETRYACK_TXRSP monitors the efficiency of using dynamic credits in the system. It does this task by measuring the request retry rate:

RETRYACK_TXRSP

Number of **RXREQ** flits dispatched. This event is a measure of the retry rate. Calculate the retry rate as follows:

$$\text{Retry rate} = \text{RETRYACK_TXRSP} / \text{RXREQFLITV_TOTAL}$$

Delays at SBSX bridges due to backpressure

To analyze the delays in SBSX bridges, CMN-700 enables you to monitor the source of backpressure.

SBSX might have requests that are ready to be sent to the downstream AXI/ACE-Lite device, but cannot send them due to backpressure from the downstream device. In this situation, SBSX holds the request in the *Receive Request Tracker* (RRT). This condition results in the RRT getting full and so the SBSX bridge cannot accept any new requests from RNs impacting system performance.

The following table contains events that monitor such backpressure from the downstream AXI or ACE-Lite device:

Table 6-3 AXI/ACE-Lite downstream monitoring events

Events	Description
ARVALID_NO_ARREADY	Number of cycles the SBSX bridge is stalled due to backpressure on AR channel.
AWVALID_NO_AWREADY	Number of cycles the SBSX bridge is stalled due to backpressure on AW channel.
WVALID_NO_WREADY	Number of cycles the SBSX bridge is stalled due to backpressure on W channel.

If a mesh is congested with many DAT or RSP flits, it might not give link credits to SBSX in timely manner. This situation can cause DAT flits for Reads or RSP flits for Writes to be stalled in SBSX. The following table describes events monitor in such cases where SBSX bridge is not able to upload DAT/RSP flits on the mesh.

Table 6-4 CHI events monitor information

Events	Description
TXDATFLITV_NO_LINKCRD	Number of cycles the TXDAT flit in SBSX bridge is waiting for link credits.
TXRSPFLITV_NO_LINKCRD	Number of cycles the TXRSP flit in SBSX bridge is waiting for link credits.

Tracker occupancy analysis

To debug performance issues, more events are provided to measure occupancy of various trackers in SBSX. These trackers include the *Request Received Tracker* (RRT), *Request Dispatch Tracker* (RDT), and *Write Data Buffers* (WDB).

Read, Write, and CMO transactions occupy RRT before they are dispatched on the AXI interface. When Read/CMO transactions are dispatched on AXI, they move from RRT to RDT. Writes remain on RRT until the write response is obtained from AXI interface and then deallocated from RRT. Knowing the occupancy of RRT and RDT independently can inform you better about the bottleneck source. In the PMU event register description section, RRT is called request tracker, while RDT is called AXI pending tracker.

The following table contains tracker occupancy information.

Table 6-5 Tracker occupancy information

Events	Description
RRT_RD_OCCUPANCY_CNT_OVFL	Read request tracker occupancy count overflow
RRT_WR_OCCUPANCY_CNT_OVFL	Write request tracker occupancy count overflow
RRT_CMO_OCCUPANCY_CNT_OVFL	CMO request tracker occupancy count overflow
WDB_OCCUPANCY_CNT_OVFL	WDB occupancy count overflow
RDT_RD_OCCUPANCY_CNT_OVFL	Read AXI pending tracker occupancy count overflow
RDT_CMO_OCCUPANCY_CNT_OVFL	CMO AXI pending tracker occupancy count overflow

6.5.3 SBSX PMU event summary

This section contains SBSX PMU event summary information.

For more information, see [*por_sbsx_pmu_event_sel* on page 3-900](#).

6.6 MTSX performance events

The MTSX implements all the SBSX performance monitoring events and also MTU-specific performance monitoring events.

Up to four MTSX events can be selected using configuration registers. Given that SBSX and MTU events are combined as MTSX events, each event should be selected in either SBSX or MTU only. If the same event is selected in both SBSX and MTU, the one selected in the SBSX is counted.

For more information about the performance monitoring in the SBSX, see [6.5 SBSX performance events on page 6-1337](#).

This section contains the following subsections:

- [6.6.1 MTSX TC performance on page 6-1341](#).
- [6.6.2 MTSX bandwidth on page 6-1341](#).

6.6.1 MTSX TC performance

You can use certain MTSX PMU events to characterize the behavior of the TC and make measurements related to the performance of the TC.

The following events can be used to characterize MTSX TC behavior:

- PMU_TC_LOOKUP_EVENT
- PMU_TC_FILL_EVENT
- PMU_TC_MISS_EVENT

You can calculate the TC miss rate according to the following equation:

$$\text{TC miss rate (\%)} = \frac{\text{PMU_TC_MISS_EVENT}}{\text{PMU_TC_LOOKUP_EVENT}} \times 100$$

Figure 6-13 TC miss rate

Certain tag requests require multiple accesses to the TC. PMU_TC_LOOKUP_EVENT counts first-time accesses, in other words TC lookup, only.

You can use PMU_TC_FILL_EVENT as an approximation of the TC usage. All TC allocations count toward this event. This event does not count updates to a line already installed in the TC.

6.6.2 MTSX bandwidth

You can use a combination of MTSX PMU events to calculate the MTSX bandwidth.

The following events can be used for MTSX bandwidth information:

- PMU_AXI_RD_REQ_EVENT
- PMU_AXI_WR_REQ_EVENT

You can calculate the actual read bandwidth seen at the MTSX from AXI using the following equation:

$$\text{Actual read bandwidth} = \frac{\text{PMU_AXI_RD_REQ_EVENT} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{frequency}$$

Figure 6-14 MTSX actual read bandwidth

You can calculate the actual write bandwidth seen at the MTSX from AXI using the following equation:

$$\text{Actual write bandwidth} = \frac{\text{PMU_AXI_WR_REQ_EVENT} \times \text{DataFltSize}}{\text{Cycles}} \times \text{frequency}$$

Figure 6-15 MTSX actual write bandwidth

MTSX PMU event summary

The MTSX implements several performance monitoring events to allow you to monitor the performance of the unit.

Both SBSX and MTU-specific PMU events are implemented in MTSX. See [por_sbsx_pmu_event_sel on page 3-900](#) for the SBSX PMU events. The following table shows the MTU-specific MTSX PMU events and their encodings.

Table 6-6 MTSX PMU event summary

Number	Name	Description
1	PMU_TC_LOOKUP_EVENT	TC lookup requests. This event measures TC accesses.
2	PMU_TC_FILL_EVENT	TC allocation (Dirty or Clean) requests. This event measures TC allocations.
3	PMU_TC_MISS_EVENT	TC misses. This event measures TC miss rate.
4	PMU_TDB_FORWARD_EVENT	Requests that received data from TDB forwarding.
5	PMU_TCQ_HAZARD_EVENT	TCQ address hazards on allocation
6	PMU_TCQ_RD_ALLOC_EVENT	Read requests allocated in TCQ, including read tag ops and write match tag ops.
7	PMU_TCQ_WR_ALLOC_EVENT	Write requests allocated in TCQ. This event counts write update tag ops.
8	PMU_TCQ_CMO_ALLOC_EVENT	CMO requests allocated in TCQ. This event counts CMOs and write+CMOs.
9	PMU_AXI_RD_REQ_EVENT	Read requests sent out on AXI. This event measures MTSX AXI read bandwidth.
10	PMU_AXI_WR_REQ_EVENT	Write requests sent out on AXI. This event measures MTSX AXI write bandwidth.
11	PMU_TCQ_OCC_CNT_OVFL_EVENT	TCQ tracker occupancy count overflow. This event measures oversubscription of the TCQ tracker.
12	PMU_TDB_OCC_CNT_OVFL_EVENT	TDB occupancy count overflow. This event measures oversubscription of the TDB.

6.7 HN-I performance events

This section contains HN-I performance event information.

This section contains the following subsections:

- [6.7.1 Bandwidth at HN-I bridges on page 6-1343](#).
- [6.7.2 Bottleneck analysis at HN-I bridges on page 6-1344](#).
- [6.7.3 HN-I PMU event summary on page 6-1346](#).
- [6.7.4 HN-P PMU events on page 6-1346](#).

6.7.1 Bandwidth at HN-I bridges

This section contains HN-I bridge bandwidth information.

The following events are used to measure bandwidth at the HN-I bridges:

- [Read bandwidth on interconnect at HN-I bridges on page 6-1343](#).
- [Write bandwidth at HN-I bridges on page 6-1343](#).
- [Total requested bandwidth at HN-I bridges on page 6-1344](#).

Read bandwidth on interconnect at HN-I bridges

This section contains information on read bandwidth on interconnect at HN-I bridges.

This event counts the number of received data flits at the HN-I and interconnect:

PMU_HNI_RXDAT Number of **RXDAT** flits received at XP from HN-I. This event is a measure of the read data bandwidth.

Calculate the actual read bandwidth as follows:

$$\text{Actual read bandwidth} = \frac{\text{PMU_HNI_RXDAT} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-16 Actual read bandwidth

————— Note ————

This event is tracked in the DTM watchpoint in the XP where the component is located.

————— Note ————

This event is tracked in the DWM, not in the HN-I design. The event is defined from the XP's perspective.

Write bandwidth at HN-I bridges

This section contains information on write bandwidth at HN-I bridges.

In a similar way to the read actual bandwidth event, this event monitors the number of data flits that the HN-I receives. Therefore this event measures the actual write bandwidth that is received from the interconnect:

PMU_HNI_TXDAT Number of **TXDAT** flits dispatched from XP to HN-I. This event is a measure of the write bandwidth.

Calculate the write bandwidth as follows:

$$\text{Actual write bandwidth} = \frac{\text{PMU_HNI_TXDAT} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-17 Actual write bandwidth

————— **Note** ————

This event is tracked in the DTM watchpoint in the XP where the component is located.

————— **Note** ————

This event is tracked in the DWM, not in the HN-I design. The event is defined from the perspective of the XP.

Total requested bandwidth at HN-I bridges

This section contains information on total requested bandwidth at HN-I bridges.

To improve efficiency when using PMU events and signals, this event combines the read and write bandwidth estimation in a single event. The PMU_HNI_TXREQ_TOTAL event monitors the number of REQ flits that an HN-I bridge receives:

PMU_HNI_TXREQ_TOTAL

Number of TXREQ flits dispatched from XP to HN-I. This event is a measure of the total request bandwidth.

Calculate the total bandwidth as follows:

$$\text{Total requested bandwidth} = \frac{\text{PMU_HNI_TXREQ} \times \text{DataFlitSize}}{\text{Cycles}} \times \text{Frequency}$$

Figure 6-18 Total requested bandwidth

————— **Note** ————

This event is tracked in the DTM watchpoint in the XP where the component is located.

————— **Note** ————

This event is tracked in the DWM, not in the HN-I design. The event is defined from the perspective of the XP.

6.7.2 Bottleneck analysis at HN-I bridges

This section contains information on bottleneck analysis at HN-I bridges.

Locations where the nodes or bridges are full can cause delays in the rest of the system. CMN-700 provides events that observe locations where the nodes or bridges are full. This feature enables you to monitor the current bottlenecks in the system, and checks multiple events in all CMN-700 components. The events monitor the following:

- The number of times the bridge is forced to retry because of the lack of dynamic credits.
- The number of times requests are serialized due to ordering requirements.
- The number of cycles the bridge is forced to stall due to backpressures.

The following events are used to measure bottlenecks at the HN-I bridges:

- [6.7 HN-I performance events on page 6-1343](#).

Request retry rate at HN-I bridges

This section contains information on the request retry rate at HN-I bridges.

RETRYACK_TXRSP monitors the efficiency of using dynamic credits in the system. It does this task by measuring the request retry rate:

RETRYACK_TXRSP

Number of **RXREQ** flits dispatched. This event is a measure of the retry rate. Calculate the retry rate as follows:

$$\text{Retry rate} = \text{RETRYACK_TXRSP} / \text{RXREQFLITV_TOTAL}$$

Delays at HN-I bridges because of ordering requirements

When requests are received at an HN-I, there are different ordering guarantees the HN-I bridge must maintain based on the source and attributes of the request.

The requests are sometimes serialized, indicating a lower than expected bandwidth at HN-I, as the following table shows.

Table 6-7 PCIe and non-PCIe RN request information

Request	Description
NONPCIE_SERIALIZED	Number of non-PCIe RN requests that are serialized.
PCIE_SERIALIZED	Number of PCIe RN requests that are serialized.

Delays at HN-I bridges because of backpressure

To analyze the delays in HN-I bridges, CMN-700 enables you to monitor the source of backpressure.

HN-I might have requests that are ready to be sent to AXI/ACE-Lite downstream, but cannot send them due to backpressure from AXI/ACE-Lite downstream. In this situation, HN-I holds the request in the RRT. As a result, the RRT gets full. Therefore, the HN-I bridge cannot accept any new requests from RNs, impacting system performance.

This table describes the events that monitor such backpressure from AXI/ACE-Lite downstream:

Table 6-8 AXI/ACE downstream events monitor information

Events	Description
ARVALID_NO_ARREADY	Number of cycles the HN-I bridge is stalled due to backpressure on AR channel.
AWVALID_NO_AWREADY	Number of cycles the HN-I bridge is stalled due to backpressure on AW channel.
WVALID_NO_WREADY	Number of cycles the HN-I bridge is stalled due to backpressure on W channel.

Even if the AXI/ACE-Lite downstream is ready to accept new requests, the HN-I bridge cannot send them downstream while the RDT is full. The lifetime of a request in the RDT depends on response latency from AXI/ACE-Lite downstream and backpressure on TXDAT channel.

This table describes the events that monitor cases where an HN-I bridge is unable to send new requests to AXI/ACE-Lite downstream:

Table 6-9 AXI/ACE downstream events monitor information (no new requests sent)

Events	Description
ARREADY_NO_ARVALID	Number of cycles the AR channel is waiting for new requests from HN-I bridge.
AWREADY_NO_AWVALID	Number of cycles the AW channel is waiting for new requests from HN-I bridge.

If the mesh is congested with many DAT Flits then there might be a delay before it gives link credits to HN-I. This delay results in the stalling of DAT flits for Reads in HN-I. This table describes events that monitor cases where an HN-I bridge is not able to upload a DAT flit on the mesh.

Table 6-10 CHI events monitor information

Events	Description
TXDATFLITV_NO_LINKCRD	Number of cycles the TXDAT flit in HN-I bridge is waiting for link credits.

Tracker occupancy analysis in HN-I

To debug performance issues, more events are provided to measure occupancy of various trackers in HN-I such as RRT, RDT, and WDB.

Read and Write transactions occupy RRT before they are dispatched on the AXI interface. When Read and Write transactions are dispatched on AXI, they move from RRT to RDT. Reads and Writes remain on RDT until all the responses are obtained from the AXI interface. The transactions are then deallocated from RDT. Knowing the occupancy of RRT and RDT independently can inform you better about the bottleneck source. In the PMU event register description section, RRT is called request tracker while RDT is called AXI pending tracker.

The following table contains tracker occupancy information:

Table 6-11 Tracker occupancy information

Events	Description
RRT_RD_OCCUPANCY_CNT_OVFL	Read occupancy count overflow event in RRT
RRT_WR_OCCUPANCY_CNT_OVFL	Write occupancy count overflow event in RRT
RDT_RD_OCCUPANCY_CNT_OVFL	Read occupancy count overflow event in RDT
RDT_WR_OCCUPANCY_CNT_OVFL	Write occupancy count overflow event in RDT
WDB_OCCUPANCY_CNT_OVFL	WDB occupancy count overflow event

6.7.3 HN-I PMU event summary

This section contains HN-I PMU event summary information.

For more information, see [por_hni_pmu_event_sel](#) on page 3-395.

6.7.4 HN-P PMU events

HN-P has dedicated RRTs, RDTs, and WDBs for requests from RN-Is and RN-Ds connected to PCIe masters. There are associated PMU events for these resources.

For more information about the associated events, see [por_hnp_pmu_event_sel](#) on page 3-397.

Reads from PCIe RN-Is and RN-Ds do not support Bypassstracetag.

6.8 DN performance events

This section contains DN performance event information.

The following table shows a summary of the DN PMU events.

Table 6-12 DN PMU event summary

Number	Description
1	Number of TLBI DVM op requests
2	Number of BPI DVM op requests
3	Number of PICI DVM op requests
4	Number of VICI DVM op requests
5	Number of DVM sync requests
6	Number of DVM op requests that were filtered using VMID filtering
7	Number of DVM op requests to RN-Ds, BPI, or PICI/VICI, that are filtered.
8	Number of retried REQs
9	Number of SNPs sent to RNs
10	Number of SNPs stalled to RNs due to lack of credits
11	DVM tracker full counter
12	DVM tracker occupancy counter

The pmu_occup1_id field in the por_dn_pmu_event_sel register is used to program the occupancy counter for specific operations types. The following table summarizes the options.

Table 6-13 Field values for pmu_occup1_id

pmu_occup1_id values	Description
0b0000	All
0b0001	DVM Ops
0b0010	DVM Syncs

————— Note —————

In HN-D, DN PMU events can be accessed only when the corresponding HN-I PMU select is 0 (NONE).

DN events can be accessed through the HN-D. The por_dn_pmu_event_sel register outputs on corresponding TXPMU output only if por_hni_pmu_event_sel bits [5], [13], [21], and [29] are set to 0. Otherwise the value on the HN-I PMU is available.

The por_hnp_pmu_event_sel register only outputs on the corresponding TXPMU output if por_hni_pmu_event_sel bits [5], [13], [21], and [29] are set to 0. Otherwise, the value on the HN-I PMU is available.

6.9 XP PMU event summary

This section contains XP PMU event summary information.

Each of the XP PMU events is associated with:

- One of six XP ports:
 - If using a mesh configuration, these ports can be East, West, North, South, device port P0, P1, P2, or P3, depending on the configuration.
 - If using a single-MXP configuration, these ports are device ports P0, P1, P2, P3, P4, and P5.
- One of four CHI channels - REQ, RSP, SNP, or DAT.
- One of six CHI channels - REQ, RSP, SNP, DAT, RSP2, or DAT2. RSP2 and DAT2 are valid when dual DAT and RSP mode is enabled.

Up to four XP PMU Events can be specified using the `por_mxp_pmu_event_sel` register. For more information about this register, see [por_mxp_pmu_event_sel on page 3-992](#).

The following table shows a summary of the XP PMU events.

Table 6-14 XP PMU event summary

Number	Name	Description
1	PMU_XP_TXFLIT_VALID	<p>Number of flits that are transmitted on a specified port and CHI channel. This event measures the flit transfer bandwidth from an XP.</p> <hr/> <p>———— Note ————</p> <p>On device ports, this event also includes link flit transfers.</p> <hr/>
2	PMU_XP_TXFLIT_STALL	Number of cycles when a flit is stalled at an XP waiting for link credits at a specified port and CHI channel. This event measures the flit traffic congestion on the mesh and at the flit download ports.

6.10 CXG performance events

This section contains CXG performance event information.

The XP considers CXRH as a single node when selecting PMU events. The PMU events for CXRA and CXHA share the four possible PMU counters for that port in the XP. Therefore a total of four events can be counted simultaneously from CXRA+CXHA. The CXHA events have "priority" over the CXRA events. If an event selection (0, 1, 2, 3) in por_cxg_ha_pmu_event_sel is nonzero (is not CXHA_PMU_EVENT_NULL), then the event selected in por_cxg_ha_pmu_event_sel is sent to the XP. If an event selection (0, 1, 2, 3) in por_cxg_ha_pmu_event_sel is zero (is CXHA_PMU_EVENT_NULL), then the event selected in por_cxg_ra_pmu_event_sel is sent to the XP.

The CXLA contains its own PMU event counters. As with XP PMU counters, you can pair CXLA PMU event counters with the global counters in a DTC. Configure the CXLA PMU event counters in por_cxla_pmuevent.

————— Note —————

To enable CXLA PMU functionality, you must configure por_cxla_pmu_config.pmu_en and por_cxg_ra_cfg_ctl.en_cxla_pmucmd_prop to 1'b1. Also configure the usual DT and PMU enables.

6.11 Occupancy and lifetime measurement using PMU events

CMN-700 has PMU events to measure the average occupancy of a tracker and measure the average lifetime of the requests in that tracker.

This event is implemented for many of the trackers in CMN-700 units (HN-F, RN-I, RN-D, HN-I, and others). The following formula measures the average occupancy and lifetime and can be applied to all the trackers where this event is supported:

Occupancy Measurement

The formula to measure the occupancy is:

$$\text{Average Occupancy (entries)} = \frac{\text{PMU_OCCUPANCY_EVENT} \ll 12}{\text{PMU_CYCLE_COUNTER}}$$

Figure 6-19 Average occupancy

For example, for RN-I RRT average occupancy, the formula is:

$$\text{Average RRT Occupancy (entries)} = \frac{\text{PMU_RNI_RRT_OCCUPANCY_EVENT} \ll 12}{\text{PMU_CYCLE_COUNTER}}$$

Figure 6-20 Average RRT occupancy

Lifetime Measurement

If a tracker supports lifetime event, the formula to measure the lifetime is:

$$\text{Average Lifetime (cycles)} = \frac{\text{PMU_OCCUPANCY_EVENT} \ll 12}{\text{PMU_NUM_TRACKER_ALLOCATIONS}}$$

Figure 6-21 Average lifetime

For example, for RN-I RRT average lifetime, the formula is:

$$\text{Average Lifetime (cycles)} = \frac{\text{PMU_RNI_RRT_OCCUPANCY} \ll 12}{\text{PMU_RNI_RRTALLOC}}$$

Figure 6-22 Average RRT lifetime

HN-F supports collecting occupancy according to the request types. The following table describes the opcode filtering types that are supported.

Table 6-15 Supported opcode filtering types

pmu_occup1_id	Opcode type
0b000	All request types
0b001	Read request types
0b010	Write request types
0b011	Atomic request types
0b100	Stash request types

When filtering is enabled, pmu_occup1_id must return to the default value to collect occupancy for all request types.

6.12 DEVEVENT

CMN-700 HN-Fs support device-specific events that are together called DEVEVENT. These events are sent along with the completion of a transaction.

Completion of a transaction can be a data response (DAT) or completion response (RSP). These events contain information regarding the transaction encountering SLC hit or miss. And it also includes information about snoops sent to resolve coherency actions. These events can be measured using watchpoints on the XP that the RN-F is connected. Refer to [5.1.1 DTM watchpoint on page 5-1302](#) for watchpoint usage.

The following table describes the DEVEVENT encodings from HN-F.

Table 6-16 DEVEVENT encodings from HN-F

Encoding	Description
2'b00	Line missed in SLC and no snoops sent.
2'b01	Line missed in SLC and directed snoop sent.
2'b10	Line missed in SLC and broadcast snoops sent.
2'b11	Line hit in SLC and no snoops sent.

Responses from other CMN-700 devices have the default 2'b00 as the DEVEVENT value.

Appendix A

Protocol feature compliance

This appendix describes the various features that CMN-700 implements from different protocol and architecture specifications.

It contains the following sections:

- [*A.1 AXI/ACE-Lite feature support* on page Appx-A-1353.](#)
- [*A.2 CHI feature support* on page Appx-A-1355.](#)
- [*A.3 CXS property support* on page Appx-A-1356.](#)
- [*A.4 CCIX property support* on page Appx-A-1357.](#)
- [*A.5 CHI feature support for CML* on page Appx-A-1358.](#)

A.1 AXI/ACE-Lite feature support

AXI/ACE-Lite provides various optional features through interface properties. CMN-700 supports some of these properties and whether a property is supported depends on the node type.

The following table shows the AXI/ACE-Lite properties that are supported by the different CMN-700 nodes with AXI/ACE-Lite interfaces.

Table A-1 AXI/ACE-Lite feature support

AXI/ACE-Lite property	Support		
	RN-I	HN-I	SBSX
Wakeup_Signals	Y	Y	Y
Check_Type	Y	Y	Y
Poison	Y	Y	Y
Trace_Signals	Y	Y	Y
Unique_ID_Support	Y	Y	Y
QoS_Accept	N	N	N
Loopback_Signals	Y	N	N
Untranslated_Transactions	N	N	N
NSAccess_Identifiers	Y	N	Y
CMO_On_Read	Y	N	Y
CMO_On_Write	Y	N	Y
Persist_CMO	Y	N	Y
Write_Plus_CMO	Y	N	Y
DVM_v8 and DVM_v8.1	Y (RN-D)	N	N
DVM_v8.4	Y (RN-D)	N	N
Coherency_Connection_Signals	Y	N	N
MPAM_Support	Y	Y	Y
Read_Interleaving_Disabled	Y	Y	Y
Read_Data_Chunking	Y	N	N
Cache_Stash_Transactions	Y	N	N
Atomic_Transactions	Y	N	N
DeAllocation_Transactions	Y ^a	N	N
WriteEvict_Transaction	N	N	N
Barrier_Transactions	N	N	N
Ordered_Write_Observation	Y	N	N
DVM_On_Read	Y (RN-D)	N	N
DVM_On_Snoop	Y (RN-D)	N	N

^a ReadOnceMakeInvalid or ReadOnceCleanInvalid

Table A-1 AXI/ACE-Lite feature support (continued)

AXI/ACE-Lite property	Support		
	RN-I	HN-I	SBSX
Max_Transaction_Size	4KB	64B for HN-I/D/T/V & Any for HN-P	64B
WriteZero	N	N	Y
Fixed_Burst	Y	N	N
Exclusive_Access	Y	N	N
Shareable_Transactions	Y	N	N
Prefetch_Transaction	N	N	Y

A.2 CHI feature support

CHI provides various optional features through interface properties. CMN-700 supports some of these properties.

The following table shows the CHI properties that are supported by CMN-700.

Table A-2 CHI feature support

CHI property	Support	Comments
Atomic_Transactions	Y	-
Cache_Stash_Transactions	Y	-
Direct_Memory_Transfer	Y	-
Direct_Cache_Transfer	Y	-
Data_Poison	Y	-
Data_Check	Y	-
CCF_Wrap_Order	N	True for most of the nodes, but not all.
Req_Addr_Width	Y	-
NodeID_Width	Y	Supported values are 7-11.
Data_Width	N	Fixed to 256
Enhanced_Features	Y	Support enabled for all enhanced features.
CleanSharedPersistSep_Request	Y	-
MPAM_Support	Y	-

A.3 CXS property support

This section provides information on CXS properties and their relevant support.

The following table contains the supported CXS properties.

Table A-3 CXS property support

CXS property	Support	Comments
TX and RX CXSDATAFLITWIDTH	256/512	CXSDATAFLITWIDTH values are always the same for CXS TX and RX.
TX and RX CXSMAXPKTPERFLIT	2/4	If TX/RX CXSDATAFLITWIDTH = 512, the value of TX/RX CXSMAXPKTPERFLIT = 4.
TX CXSCONTINUOUSDATA	True	-
RX CXSCONTINUOUSDATA	False	-
TX and RX CXSErrorFullPkt	True	-
TX and RX CXSDATACHECK	None	-
TX and RX CXSREPLICATION	None	-

A.4 CCIX property support

This section provides CCIX information for CML support.

The following table contains CCIX property settings for CML.

Table A-4 CCIX property settings for CML

Property	Permitted values	Support
NoCompAck	True, False	CXG = False CCG = Both (True is the Default)
PartialCacheStates	True, False	False
CacheLineSize	64B, 128B	64B
AddrWidth	48b, 52b, 56b, 60b, 64b	48b/52b
PktHeader	Compatible, Optimized	CXG = Both CCG = Optimized
MaxPacketSize	128B, 256B, 512B	All (applicable to CXG only)
NoMessagePack	True, False	CXG = Both (True is the Default) CCG = False

— Restriction —

The following CCIX features are not supported:

- Concurrent memory expansion.
- Snoop chaining outbound.
- CCIX snoop multicast (inbound and outbound).
- Snoop broadcast outbound.

A.5 CHI feature support for CML

This section provides CHI information for CML support.

The following table contains CHI support for CML settings.

Table A-5 CHI support for CML

CHI Feature	CML Support		Comments
	Local	Remote	
Coherency	Yes	Yes	-
Ordering	Yes	Yes	-
Atomics	Yes	Yes	-
Exclusive Accesses	Yes	Yes	Refer to 2.9.2 Exclusive accesses on page 2-196 for more details on CML handling of Exclusive Accesses.
Cache Stashing	Yes	SMP Mode Only*	Remote Support: Stash* request are sent to the remote chip over CCIX. Stash*ID fields are not passed through along with the request. SnpStash* are not supported over CCIX link.
DVM Operations	Yes	SMP Mode Only	-
Error Handling			
- Response Error	Yes	Yes	-
- Data Check	Yes	No	-
- Poison	Yes	Yes	Mandatory for CMN-700.
QoS			
- Request	Yes	Yes	-
- Snoop	Yes	No	-
Data Return from Shared Clean	Yes	No	-
<i>Direct Cache Transfer (DCT)</i>	Yes	No	Local support includes local RN-F sending data directly to CCIX gateway block.
<i>Direct Memory Transfer (DMT)</i>	Yes	No	Local support includes local SN-F sending data directly to CCIX gateway block.
I/O Deallocation Transactions	Yes	Yes	-
CleanSharedPersist CMO	Yes	Yes	-
CleanSharedPersistSep CMO	Yes	SMP Mode Only	-
Prefetch Target	Yes	No	Remote Support: Not supported over CCIX
Trace Tag	Yes	SMP Mode Only	-
System Coherency Interface (SYSCOREQ and SYCOACK)	Yes	Yes (using s/w bits)	-
Partial Cache State	Yes	No	CXRA, inside CML block, does not accept the following requests and responses: WriteBackPtl, WriteCleanPtl, SnpRespDataPtl*
Streaming and Optimized Streaming of Ordered Writes	Yes	SMP Mode Only	Remote Support: In SMP mode, High BW streaming of ordered writes from PCIe RN-I/D can be enabled through CCG
MPAM	Yes	SMP Mode Only	-

Table A-5 CHI support for CML (continued)

CHI Feature	CML Support		Comments
	Local	Remote	
CHI-A RN-F	No	No	Remote Support: CXRA relies on the DoNotGoToSD field of the CHI Snoop to map CCIX Snoop request SnpToSC and therefore does not support CHI-A RN-Fs. Because of this, CHI-A RN-Fs are also not supported for local traffic.
CHI Ops	Yes	SMP Mode Only	Remote Support: Both Exclusive and Non-exclusive
StashOnceSep	Yes	SMP Mode Only	-
Write*Zero	Yes	SMP Mode Only	-
Read PerferUnique	Yes	SMP Mode Only	-
W+CMO	Yes	SMP Mode Only*	* Sent as separate Write and CMO Ops over CCIX
SLC Replacement	Yes	SMP Mode Only	-
Deep Attribute	Yes	SMP Mode Only	-
SnpQuery	Yes	SMP Mode Only	-
SnpPerferUnique	Yes	SMP Mode Only	-

CML Requirement

Each CML Port requires a minimum of one request and one data credit more than the total number of reservations. This number is enabled through the CXRA configuration control register (*por_cxg_ra_cfg_ctl*).

This requirement applies to each enabled CCIX link at a given CCIX Port. For example, by default all the reservations are enabled in the SMP mode. Therefore, a minimum of four request and four data credits are required to be granted per CCIX link. These credits are used by certain traffic types, such as QoS-15, to make progress in a loaded system. See configuration register *por_cxg_ra_cfg_ctl* on page 3-336 for more details.

Appendix B

Signal descriptions

This appendix describes the external I/O signals that CMN-700 implements for connection to other hardware in the system.

It contains the following sections:

- [*B.1 About the signal descriptions* on page Appx-B-1361.](#)
- [*B.2 Clock and reset signals* on page Appx-B-1362.](#)
- [*B.3 CHI interface signals* on page Appx-B-1363.](#)
- [*B.4 ACE-Lite and AXI Interface signals* on page Appx-B-1369.](#)
- [*B.5 APB interface signals* on page Appx-B-1385.](#)
- [*B.6 Device population signals* on page Appx-B-1386.](#)
- [*B.7 Debug, trace, and PMU interface signals* on page Appx-B-1387.](#)
- [*B.8 DFT and MBIST interface signals* on page Appx-B-1389.](#)
- [*B.9 Clock management signals* on page Appx-B-1394.](#)
- [*B.10 Power management signals* on page Appx-B-1395.](#)
- [*B.11 Interrupt and event signals* on page Appx-B-1396.](#)
- [*B.12 Configuration input signals* on page Appx-B-1397.](#)
- [*B.13 Processor event interface signals* on page Appx-B-1398.](#)
- [*B.14 CGL interface signals* on page Appx-B-1399.](#)
- [*B.15 CXLA configuration interface signals* on page Appx-B-1405.](#)

B.1 About the signal descriptions

CMN-700 signals are composed of a base name along with identifiers that indicate unique product configuration.

Because there are multiple identical interfaces in CMN-700, the signal names that this appendix describes are only root names, in many cases. The actual signal name includes a port-specific identifier suffix.

The system configuration determines which of the signals are used in a particular system.

————— **Note** —————

Unless specified otherwise, CMN-700 signals are active-HIGH.

B.2 Clock and reset signals

The following table shows the CMN-700 clock and reset signals.

Table B-1 CMN-700 clock and reset signals

Signal	Type	Description	Connection information
GCLK0	Input	Primary CMN-700 clock input. This clock signal is always present.	Connect to global clock for CMN-700.
GCLK1	Input	Primary CMN-700 clock input. This clock is only present if you divide the mesh into four asynchronous clock domains.	Connect to global clock for CMN-700.
GCLK2	Input	Primary CMN-700 clock input. This clock is only present if you divide the mesh into four asynchronous clock domains.	Connect to global clock for CMN-700.
GCLK3	Input	Primary CMN-700 clock input. This clock is only present if you divide the mesh into four asynchronous clock domains.	Connect to global clock for CMN-700.
nSRESET	Input	CMN-700 reset, active-LOW	Connect to global reset for CMN-700.

B.3 CHI interface signals

CMN-700 uses channels that form an inbound and outbound CHI interface for each device using signals that form each channel in a specific interface.

The *AMBA® 5 CHI Architecture Specification* defines four channels:

- *Request* (REQ).
- *Response* (RSP).
- *Snoop* (SNP).
- *Data* (DAT).

Note

All signal names in this section are only a root name, **RootName**. CMN-700 interfaces use **RootName** within a more fully specified signal name as follows:

- CMN-700 interface signal name == **RootName_NID#**, where # is the node ID corresponding to the specific interface.

This section contains the following subsections:

- [B.3.1 Per-device interface definition](#) on page Appx-B-1363.
- [B.3.2 Per-channel interface signals](#) on page Appx-B-1364.
- [B.3.3 Non-channel-specific interface signals](#) on page Appx-B-1367.
- [B.3.4 RSVDC signal description](#) on page Appx-B-1368.

B.3.1 Per-device interface definition

Each CHI device included in a CMN-700 system has distinct functionality, and the requirements and configuration of its respective CHI interfaces differ.

The requirements and configuration for the CHI interfaces are as follows:

External RN-F interface

The RN-F interface consists of a request channel, snoop channel, and two response channels, one in each direction, as the following figure shows.

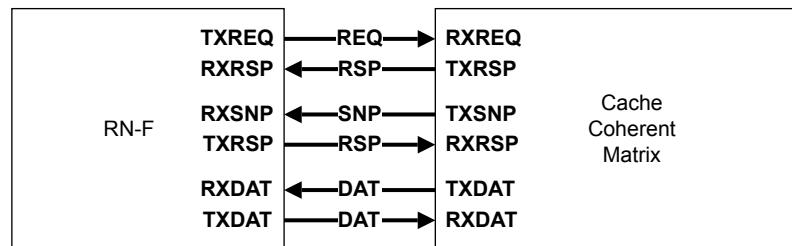


Figure B-1 External RN-F interface

It also has two data channels, one in each direction, for data transfers. CMN-700 receives request messages from the RN-F and sends responses to it. In addition, CMN-700 sends snoop messages to the RN-F and receives snoop response messages.

External SN-F interface

The SN-F interface consists of a request channel and a response channel as the following figure shows.

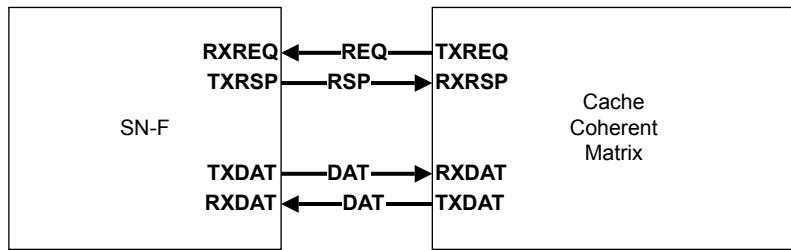


Figure B-2 External SN-F interface

It also has two data channels, one in each direction, for data transfers. The SN-F receives request messages from CMN-700 and returns response messages.

B.3.2 Per-channel interface signals

For communication between devices, each channel includes a *Transmit* (TX) and a *Receive* (RX) port, with various interface signals traveling from TX to RX.

————— Note —————

Connection of CHI interfaces between two devices requires cross-coupling of the **TX*** and **RX*** signals between the two devices, as required by the CHI architecture.

The following table shows the Transmit Request channel signals.

Table B-2 Transmit Request channel signals

Signal	Type	Description	Connection information
TXREQFLITPEND	Output	Transmit Request Early Flit Valid hint	Connect to RXREQFLITPEND of the corresponding CHI device, if populated.
TXREQFLITV	Output	Transmit Request Flit Valid	Connect to RXREQFLITV of the corresponding CHI device, if populated.
TXREQFLIT[n:0]^b	Output	Transmit Request Flit	Connect to RXREQFLIT of the corresponding CHI device, if populated.
TXREQLCRDV	Input	Transmit Request channel link layer credit	Connect to RXREQLCRDV of the corresponding CHI device, if populated, otherwise tie LOW.

The following table shows the Transmit Response channel signals.

Table B-3 Transmit Response channel signals

Signal	Type	Description	Connection information
TXRSPFLITPEND	Output	Transmit Response Early Flit Valid hint	Connect to RXRSPFLITPEND of the corresponding CHI device, if populated.
TXRSPFLITV	Output	Transmit Response Flit Valid	Connect to RXRSPFLITV of the corresponding CHI device, if populated.

^b The value of n is configuration-dependent.

Table B-3 Transmit Response channel signals (continued)

Signal	Type	Description	Connection information
TXRSPFLIT[n:0]^b	Output	Transmit Response Flit	Connect to RXRSPFLIT of the corresponding CHI device, if populated.
TXRSPLCRDV	Input	Transmit Response channel link layer credit	Connect to RXRSPFLIT of the corresponding CHI device, if populated, otherwise tie LOW.

The following table shows the Transmit Snoop channel signals.

Table B-4 Transmit Snoop channel signals

Signal	Type	Description	Connection information
TXSNPFLITPEND	Output	Transmit Snoop Early Flit Valid hint	Connect to RXSNPFLITPEND of the corresponding CHI device, if populated.
TXSNPFLITV	Output	Transmit Snoop Flit Valid	Connect to RXSNPFLITV of the corresponding CHI device, if populated.
TXSNPFLIT[n:0]^b	Output	Transmit Snoop Flit	Connect to RXSNPFLIT of the corresponding CHI device, if populated.
TXSNPLCRDV	Input	Transmit Snoop channel link layer credit	Connect to RXSNPLCRDV of the corresponding CHI device, if populated, otherwise tie LOW.

The following table shows the Transmit Data channel signals.

Table B-5 Transmit Data channel signals

Signal	Type	Description	Connection information
TXDATFLITPEND	Output	Transmit Data Early Flit Valid hint	Connect to RXDATFLITPEND of the corresponding CHI device, if populated.
TXDATFLITV	Output	Transmit Data Flit Valid	Connect to RXDATFLITV of the corresponding CHI device, if populated.
TXDATFLIT[n:0]^b	Output	Transmit Data Flit	Connect to RXDATFLIT of the corresponding CHI device, if populated.
TXDATLCRDV	Input	Transmit Data channel link layer credit	Connect to RXDATLCRDV of the corresponding CHI device, if populated, otherwise tie LOW.

The following table shows the Receive Request channel signals.

Table B-6 Receive Request channel signals

Signal	Type	Description	Connection information
RXREQFLITPEND	Input	Receive Request Early Flit Valid hint	Connect to TXREQFLITPEND of the corresponding CHI device, if populated, otherwise tie LOW.
RXREQFLITV	Input	Receive Request Flit Valid	Connect to TXREQFLITV of the corresponding processor, if populated, otherwise tie LOW.

Table B-6 Receive Request channel signals (continued)

Signal	Type	Description	Connection information
RXREQFLIT[n:0]^b	Input	Receive Request Flit	Connect to TXREQFLIT of the corresponding CHI device, if populated, otherwise tie LOW.
RXREQLCRDV	Output	Receive Request channel link layer credit	Connect to TXREQLCRDV of the corresponding CHI device, if populated.

The following table shows the Receive Response channel signals.

Table B-7 Receive Response channel signals

Signal	Type	Description	Connection information
RXRSPFLITPEND	Input	Receive Response Early Flit Valid hint	Connect to TXRSPFLITPEND of the corresponding CHI device, if populated, otherwise tie LOW.
RXRSPFLITV	Input	Receive Response Flit Valid	Connect to TXRSPFLITV of the corresponding processor, if populated, otherwise tie LOW.
RXRSPFLIT[n:0]^b	Input	Receive Response Flit	Connect to TXRSPFLIT of the corresponding CHI device, if populated, otherwise tie LOW.
RXRSPLCRDV	Output	Receive Response channel link layer credit	Connect to TXRSPLCRDV of the corresponding CHI device, if populated.

The following table shows the Receive Snoop channel signals.

Table B-8 Receive Snoop channel signals

Signal	Type	Description	Connection information
RXSNPFLITPEND	Input	Receive Snoop Early Flit Valid hint	Connect to TXSNPFLITPEND of the corresponding CHI device, if populated, otherwise tie LOW.
RXSNPFLITV	Input	Receive Snoop Flit Valid	Connect to TXSNPFLITV of the corresponding processor, if populated, otherwise tie LOW.
RXSNPFLIT[n:0]^b	Input	Receive Snoop Flit	Connect to TXSNPFLIT of the corresponding CHI device, if populated, otherwise tie LOW.
RXSNPLCRDV	Output	Receive Snoop channel link layer credit	Connect to TXSNPLCRDV of the corresponding CHI device, if populated.

The following table shows the Receive Data channel signals.

Table B-9 Receive Data channel signals

Signal	Type	Description	Connection information
RXDATFLITPEND	Input	Receive Data Early Flit Valid hint	Connect to TXDATFLITPEND of the corresponding CHI device, if populated, otherwise tie LOW.
RXDATFLITV	Input	Receive Data Flit Valid	Connect to TXDATFLITV of the corresponding processor, if populated, otherwise tie LOW.

Table B-9 Receive Data channel signals (continued)

Signal	Type	Description	Connection information
RXDATFLIT[n:0]^b	Input	Receive Data Flit	Connect to TXDATFLIT of the corresponding CHI device, if populated, otherwise tie LOW.
RXDATLCRDV	Output	Receive Data channel link layer credit	Connect to TXDATLCRDV of the corresponding CHI device, if populated.

B.3.3 Non-channel-specific interface signals

Every transmit and receive link layer interface includes extra signals that exist only at the interface level and are not channel specific.

The following table shows the LinkActive interface signals.

Table B-10 Receive LinkActive interface signals

Signal	Type	Description	Connection information
RXLINKACTIVEREQ	Input	Receive channel LinkActive request from adjacent transmitter device	Connect to TXLINKACTIVEREQ of the corresponding CHI device, if populated, otherwise tie LOW.
RXLINKACTIVEACK	Output	Receive channel LinkActive acknowledgment to adjacent transmitter device	Connect to TXLINKACTIVEACK of the corresponding CHI device, if populated.
TXLINKACTIVEREQ	Output	Transmit channel LinkActive request to adjacent receiver device	Connect to RXLINKACTIVEREQ of the corresponding CHI device, if populated.
TXLINKACTIVEACK	Input	Transmit channel LinkActive acknowledgment from adjacent receiver device	Connect to RXLINKACTIVEACK of the corresponding CHI device, if populated, otherwise tie LOW.

The following table shows the SACTIVE interface signals.

Table B-11 SACTIVE interface signals

Signal	Type	Description	Connection information
RXSACTIVE	Input	Indication from the adjacent CHI device that it has one or more outstanding protocol-layer transactions. RXSACTIVE must remain asserted throughout the lifetime of the transaction.	Connect to TXSACTIVE of the corresponding CHI device.
TXSACTIVE	Output	Indication to the adjacent CHI device that CMN-700 has one or more outstanding protocol-layer transactions. TXSACTIVE remains asserted throughout the lifetime of the transaction.	Connect to RXSACTIVE of the corresponding CHI device.

The following table shows the hardware coherency interface signals.

Table B-12 Hardware coherency interface signals

Signal	Type	Description	Connection information
SYSCOREQ	Input	Request to enter CHI coherence domain when asserted and to exit the CHI coherence domain when deasserted. SYSCOREQ and SYSSCOACK implement a four-phase handshake protocol.	Connect to SYSCOREQ of corresponding CHI device, if populated, otherwise tie LOW.
SYSSCOACK	Output	Acknowledge CHI coherence domain entry/exit request	Connect to SYSSCOACK of corresponding CHI device, if populated.

B.3.4 RSVDC signal description

This section describes sub-fields of RSVDC fields in DAT channel.

Table B-13 DAT.RSVDC subfields description

Signal	Type	Description	Connection information
BASE[BASE_WIDTH-1:0]	Input/Output	Base	BASE_WIDTH is 4
METADATA[METADATA_WIDTH-1:0]	Input/Output	METADATA	METADATA_WIDTH controlled by RSVDC_METADATA_WIDTH . This field does not exist if RSVDC_METADATA_MODE_EN = 0

————— Note ————

The above sub-fields are fully packed in the same order as represented in the table to create **DAT RSVDC** field where **BASE** sub-field represent LSB side.

B.4 ACE-Lite and AXI Interface signals

CMN-700 interfaces use **RootName** as the signal name within a more fully specified convention.

————— Note —————

All signal names in this section consist of a root name, **RootName**. CMN-700 interfaces use **RootName** within a more fully specified signal name as follows:

- CMN-700 ACE-Lite and AXI interface signal name == **RootName_[S|M]<#a>_NID#b**, where:
 - S|M** Defines either a slave or master interface.
 - #_a Defines an optional interface identifier for a node that can support multiple AMBA interfaces.
 - #_b Defines the node ID corresponding to the specific interface.

Multi-bit signals append the bit-range identifier included in the **RootName** to the end of the full signal name.

This section contains the following subsections:

- [B.4.1 ACE-Lite-with-DVM slave interface signals](#) on page Appx-B-1369.
- [B.4.2 AXI/ACE-Lite master interface signals](#) on page Appx-B-1376.
- [B.4.3 Calculating the SBSX AxID signal widths](#) on page Appx-B-1380.
- [B.4.4 HN-I and HN-P AxID signal properties and encodings](#) on page Appx-B-1381.
- [B.4.5 A4S signals](#) on page Appx-B-1382.

B.4.1 ACE-Lite-with-DVM slave interface signals

This interface is present as the ACE-Lite-with-DVM slave port for an RN-D bridge. The signal descriptions show which signals specific to DVM functionality are not present in an ACE-Lite interface without DVM.

The following table shows the clock and power management signals.

Table B-14 Clock and power management signals

Signal	Type	Description	Connection information
ACLKEN_S	Input	AXI bus clock enable	Connect to clock enable logic. Tie HIGH if RN-I port is unused.
ACWAKEUP_S	Output	Indication that the interconnect is starting a transaction that is being sent to the DVM master (SMMU).	Connect to corresponding master device, if populated.
AWAKEUP_S	Input	Indication that the master is starting a transaction that is being sent to the interconnect.	Connect to corresponding master device, if populated, otherwise tie LOW.
RNID_SAM_STALL_DIS	Input	Disables RN SAM programming stall for specified RN.	Tie HIGH if boot programming, including RN SAM, is done through this RN-I port. Otherwise, tie LOW.
SYSCOREQ_S	Input	Request to enter DVM domain when asserted and to exit the DVM domain when deasserted. SYSCOREQ and SYSCOACK implement a four-phase handshake protocol.	Connect to corresponding master device. Tie LOW if master is not populated or does have port.
SYSCOACK_S	Output	Acknowledge for DVM domain entry or exit.	Connect to corresponding master device, if populated.

The following table shows the Write Address channel signals.

Table B-15 Write Address channel signals

Signal	Type	Description	Connection information
AWREADY_S	Output	Write address ready.	Connect to corresponding master device, if populated.
AWVALID_S	Input	Write address valid.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWID_S[10:0]	Input	Write address ID.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWADDR_S[n:0]^c	Input	Write address.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWLEN_S[7:0]	Input	Write burst length.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWSIZE_S[2:0]	Input	Write burst size.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWBURST_S[1:0]	Input	Write burst type.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWLOCK_S	Input	Write lock type.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWCACHE_S[3:0]	Input	Write memory type.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWUSER_S[n:0]	Input. Where n = (REQ_RSVDC_WIDTH +METADATA_WIDTH (if enabled) + 2 (for DownStreamCache) - 1).	User-defined signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWPROT_S[2:0]	Input	Write protection type.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWQOS_S[3:0]	Input	Write <i>Quality of Service</i> (QoS) identifier.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWSNOOP_S[3:0]	Input	Write transaction type.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWDOMAIN_S[1:0]	Input	Write Shareability domain.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWATOP_S[5:0]	Input	Atomic operation.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWSTASHNID_S[10:0]	Input	Indicates the node identifier of the physical interface that is the target interface for the cache stash operation.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWSTASHNIDEN_S	Input	When asserted, indicates that the AWSTASHNID signal is valid and should be used.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWSTASHLPID_S[4:0]	Input	Indicates the logical processor subunit that is associated with the physical interface that is the target for the cache stash operation.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWSTASHLPIDEN_S	Input	When asserted, indicates that the AWSTASHLPID signal is enabled and should be used.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWTRACE_S	Input	Trace signal that is associated with the AW Write Address channel.	Connect to corresponding master device, if populated, otherwise tie LOW.

^c The value of n is configuration-dependent.

Table B-15 Write Address channel signals (continued)

Signal	Type	Description	Connection information
AWLOOP_S[1:0]	Input	Loopback signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWMPAM_S[10:0]	Input	MPAM signal. AWMPAM[0] MPAM_NS Security indicator, default = AWPROT[1] . AWMPAM[9:1] PARTID Partition identifier, default = 0x000 . AWMPAM[10] PMG Performance monitor group, default = 0b0 .	Connect to corresponding master device, if populated, otherwise tie LOW.
AWIDUNQ_S	Input	Unique ID indicator signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWNSAID_S[3:0]	Input	Non-secure Access Identifier signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
AWCMO_S[1:0]	Input	Write address channel CMO indicator.	Connect to corresponding master device, if populated, otherwise tie LOW.

The following table shows the Write Data channel signals.

Table B-16 Write Data channel signals

Signal	Type	Description	Connection information
WREADY_S	Output	Write data ready.	Connect to corresponding master device, if populated.
WVALID_S	Input	Write data valid.	Connect to corresponding master device, if populated, otherwise tie LOW.
WDATA_S[n:0]^c	Input	Write data.	Connect to corresponding master device, if populated, otherwise tie LOW.
WSTRB_S[d:0]^d	Input	Write byte lane strobes.	Connect to corresponding master device, if populated, otherwise tie LOW.
WLAST_S	Input	Write data last transfer indication.	Connect to corresponding master device, if populated, otherwise tie LOW.

^c The value of d = (((n + 1) / 8) - 1).

Table B-16 Write Data channel signals (continued)

Signal	Type	Description	Connection information
WUSER_S[u:0]^e	Input	WUSER_S[0] is WDATACHK valid. If META_DATA_EN = 1, tie WUSER_S[u:1] to 0.	Connect to corresponding master device, if populated, otherwise tie LOW.
WTRACE_S	Input	Trace signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
WPOISON_S[p:0]^f	Input	Poison signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
WDATACHK_S[d:0]^d	Input	Data check signal.	Connect to corresponding master device, if populated, otherwise tie LOW.

The following table shows the Write Response channel signals.

Table B-17 Write Response channel signals

Signal	Type	Description	Connection information
BREADY_S	Input	Write response ready.	Connect to corresponding master device, if populated, otherwise tie LOW.
BVALID_S	Output	Write response valid.	Connect to corresponding master device, if populated.
BID_S[10:0]	Output	Write response ID.	
BRESP_S[1:0]	Output	Write response.	
BUSER_S[3:0]	Output	User response signal	
BTRACE_S	Output	Trace signal.	Connect to corresponding master device, if populated.
BLOOP_S[1:0]	Output	Loopback signal.	Connect to corresponding master device, if populated.
BIDUNQ_S	Output	Unique ID indicator signal.	Connect to corresponding master device, if populated.

The following table shows the Read Address channel signals.

^e The value of u depends on various parameters:

- u = 0 if META_DATA_EN = 0.
- u = 24 if META_DATA_EN = 1 and the AXI data bus is 512 bits wide.
- u = 12 if META_DATA_EN = 1 and the AXI data bus is 128 bits wide.

^f The value of p = ceil(DATA_WIDTH / 64) - 1.

Table B-18 Read Address channel signals

Signal	Type	Description	Connection information
ARREADY_S	Output	Read address ready.	Connect to corresponding master device, if populated.
ARVALID_S	Input	Read address valid.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARID_S[10:0]	Input	Read address ID.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARADDR_S[n:0]^c	Input	Read address.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARLEN_S[7:0]	Input	Read burst length.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARSIZE_S[2:0]	Input	Read burst size.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARBURST_S[1:0]	Input	Read burst type.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARLOCK_S	Input	Read lock type.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARCACHE_S[3:0]	Input	Read cache type.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARUSER_S[n:0]	Input. Where n = (REQ_RSVDC_WIDTH-1).	User-defined signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARPROT_S[2:0]	Input	Read protection type.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARQOS_S[3:0]	Input	Read QoS value.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARSNOOP_S[3:0]	Input	Read transaction type.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARDOMAIN_S[1:0]	Input	Read Shareability domain.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARTRACE_S	Input	Trace signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARLOOP_S[1:0]	Input	Loopback signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARMPAM_S[10:0]	Input	MPAM signal. ARMPAM[0] MPAM_NS Security indicator, default = ARPROT[1] . ARMPAM[9:1] PARTID Partition identifier, default = 0x000. ARMPAM[10] PMG Performance monitor group, default = 0b0.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARIDUNQ_S	Input	Unique ID indicator signal.	Connect to corresponding master device, if populated, otherwise tie LOW.

Table B-18 Read Address channel signals (continued)

Signal	Type	Description	Connection information
ARCHUNKEN_S	Input	Chunk enable signal. If asserted, read data for this transaction can be returned out of order, in 128-bit chunks.	Connect to corresponding master device, if populated, otherwise tie LOW.
ARNSAID_S[3:0]	Input	Non-secure Access Identifier signal.	Connect to corresponding master device, if populated, otherwise tie LOW.

The following table shows the Read Data channel signals.

Table B-19 Read Data channel signals

Signal	Type	Description	Connection information
RREADY_S	Input	Read data ready.	Connect to corresponding master device, if populated, otherwise tie LOW.
RVALID_S	Output	Read data valid.	Connect to corresponding master device, if populated.
RID_S[10:0]	Output	Read data ID.	Connect to corresponding master device, if populated.
RDATA_S[n:0] ^c	Output	Read data.	Connect to corresponding master device, if populated.
RRESP_S[1:0]	Output	Read data response.	Connect to corresponding master device, if populated.
RLAST_S	Output	Read data last transfer indication.	Connect to corresponding master device, if populated.
RUSER_S[x:0] ^g	Output	RUSER_S[0] is RDATACHK valid signal	Connect to corresponding master device, if populated.
RTRACE_S	Output	Trace signal.	Connect to corresponding master device, if populated.
RPOISON_S[p:0] ^f	Output	Poison signal.	Connect to corresponding master device, if populated.
RDATACHK_S[d:0] ^d	Output	Data check signal.	Connect to corresponding master device, if populated.
RLOOP_S[2:0]	Output	Loopback signal.	Connect to corresponding master device, if populated.
RIDUNQ_S	Output	Unique ID indicator signal.	Connect to corresponding master device, if populated.
RCHUNKV_S	Output	If asserted, RCHUNKNUM and RCHUNKSTRB are valid for this transfer.	Connect to corresponding master device, if populated.

^g The value of x depends on various parameters:

- x = 0 if META_DATA_EN = 0.
- x = 24 if META_DATA_EN = 1 and the AXI data bus is 512 bits wide.
- x = 12 if META_DATA_EN = 1 and the AXI data bus is 256 bits wide.
- x = 6 if META_DATA_EN = 1 and the AXI data bus is 128 bits wide.

Table B-19 Read Data channel signals (continued)

Signal	Type	Description	Connection information
RCHUNKNUM_S	Output	Indicates the number of chunks being transferred. Chunks are numbered incrementally from zero, according to the data width and base address of the transaction.	Connect to corresponding master device, if populated.
RCHUNKSTRB_S	Output	Indicates which part of read data is valid for this transfer. Each bit corresponds to 128 bits of data. RCHUNKSTRB[0] Corresponds to RDATA[127:0] . RCHUNKSTRB[1] Corresponds to RDATA[255:128] .	Connect to corresponding master device, if populated.

The following table shows the Snoop Address channel signals. These signals are not included in an ACE-Lite interface without DVM.

Table B-20 Snoop Address channel signals

Signal	Type	Description	Connection information
ACREADY_S	Input	Snoop address ready.	Connect to corresponding master device, if populated, otherwise tie LOW.
ACVALID_S	Output	Snoop address valid.	Connect to corresponding master device, if populated.
ACADDR_S[n:0] ^c	Output	Snoop address.	
ACSNOOP_S[3:0]	Output	Snoop transaction type.	
ACPROT_S[2:0]	Output	Snoop protection type.	
ACVMIDEXT[3:0]	Output	Snoop Address VMID Extension.	
ACTRACE	Output	Snoop address trace.	

The following table shows the Snoop Response channel signals. These signals are not included in an ACE-Lite interface without DVM.

Table B-21 Snoop Response channel signals

Signal	Type	Description	Connection information
CRREADY_S	Output	Snoop response ready.	Connect to corresponding master device, if populated.
CRVALID_S	Input	Snoop response valid.	Connect to corresponding master device, if populated, otherwise tie LOW.
CRRESP_S[4:0]	Input	Snoop response.	
CRTTRACE	Input	Snoop response trace.	

————— Note —————

WUSER_S[0] acts as a **WDATACHK** valid signal when **DATACHECK_EN** parameter is enabled.

- If **WUSER_S[0]=0**, the RN-I or RN-D synthesizes the correct **WDATACHK** value before sending it on CHI write request.
- If **WUSER_S[0]=1**, the RN-I or RN-D uses **WDATACHK** pin value to drive on CHI write request.

If the DATACHECK_EN parameter is disabled, the **WUSER_S[0]** input is ignored.

————— Note —————

RUSER_S[0] acts as an **RDATACHK** valid signal. Since the RN-I or RN-D always drives the **RDATACHK** value, **RUSER_S[0]** is set to 1 when DATACHECK_EN parameter is enabled.

If DATACHECK_EN parameter is disabled, **RUSER_S[0]** output is set to 0.

B.4.2 AXI/ACE-Lite master interface signals

HN-I and SBSX have an AXI/ACE-Lite master interface.

The following table shows the clock enable signals.

Table B-22 Clock enable signals

Signal	Type	Description	Connection information
ACLKEN_M	Input	AXI Master bus clock enable signal.	Connect to clock-enable logic.
AWAKEUP_M	Output	Indicates that CMN-700 is starting an AXI transaction.	Connect to corresponding slave device, if populated.

The following table shows the Write Address channel signals.

Table B-23 Write Address channel signals

Signal	Type	Description	Connection information
AWREADY_M	Input	Write address ready.	Connect to corresponding slave device, if populated, otherwise tie LOW.
AWVALID_M	Output	Write address valid.	Connect to corresponding slave device, if populated.
AWID_M[x:0]^h	Output	Write address ID.	
AWADDR_M[n:0]ⁱ	Output	Write address.	
AWLEN_M[7:0]	Output	Write burst length.	
AWSIZE_M[2:0]	Output	Write burst size.	
AWBURST_M[1:0]	Output	Write burst type.	
AWLOCK_M	Output	Write lock type.	
AWCACHE_M[3:0]	Output	Write cache type.	
AWUSER_M[n:0]^j	Output	User signal.	
AWPROT_M[2:0]	Output	Write protection type.	
AWQOS_M[3:0]	Output	Write QoS value.	
AWSNOOP_M[3:0]	Output	Shareable write transaction type.	
AWDOMAIN_M[1:0]	Output	Write Shareability domain.	
AWTRACE_M	Output	-	

^h For HN-I, x = 10. For HN-P, x = 19. For SBSX, x = 23. For more information, see [B.4.3 Calculating the SBSX AxID signal widths](#) on page Appx-B-1380.

ⁱ The value of n is configuration-dependent.

^j For HN-I, HN-D, HN-P, and SBSX, n = REQ_RSVDC_WIDTH.

Table B-23 Write Address channel signals (continued)

Signal	Type	Description	Connection information
AWMPAM_M[10:0]	Output	MPAM signal. AWMPAM[0] MPAM_NS Security indicator, default = AWPROT[1] . AWMPAM[9:1] PARTID Partition identifier, default = 0x000 . AWMPAM[10] PMG Performance monitor group, default = 0b0 .	Connect to corresponding slave device, if populated.
AWIDUNQ_M	Output	Unique ID indicator signal.	Connect to corresponding slave device, if populated.
AWNSAID_M[3:0]	Output	Non-secure Access Identifier signal.	Connect to corresponding slave device, if populated.
AWCMO_M[1:0]	Output	Type of CMO. This signal is only present on SBSX. It is not present on HN-I.	Connect to corresponding slave device, if populated.

The following table shows the Write Data channel signals.

————— Note —————

WDATA is configurable to 128 bits or 256 bits. **WSTRB** scales accordingly.

Table B-24 Write Data channel signals

Signal	Type	Description	Connection information
WREADY_M	Input	Write data ready.	Connect to corresponding slave device, if populated, otherwise tie LOW.
WVALID_M	Output	Write data valid.	Connect to corresponding slave device, if populated.
WDATA_M[n:0]ⁱ	Output	Write data.	Connect to corresponding slave device, if populated.
WSTRB_M[n:0]ⁱ	Output	Write byte lane strobes.	Connect to corresponding slave device, if populated.
WLAST_M	Output	Write data last transfer indication.	Connect to corresponding slave device, if populated.
WUSER_M[u:0]^k	Output	WUSER_M[0] is WDATACHK valid signal	Connect to corresponding slave device, if populated.
WPOISON_M[p:0]^l	Output	Poison signal.	Connect to corresponding master device, if populated, otherwise tie LOW.

^k The value of u depends on various parameters:

- $u = 0$ if META_DATA_EN = 0.
- $u = 24$ if META_DATA_EN = 1 and the AXI data bus is 512 bits wide.
- $u = 12$ if META_DATA_EN = 1 and the AXI data bus is 256 bits wide.
- $u = 6$ if META_DATA_EN = 1 and the AXI data bus is 128 bits wide.

^l The value of p = $((n + 1) / 64) - 1$.

Table B-24 Write Data channel signals (continued)

Signal	Type	Description	Connection information
WDATACHK_M[d:0] ^m	Output	Data check signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
WTRACE_M	Output	Trace signal.	Connect to corresponding master device, if populated, otherwise tie LOW.

The following table shows the Write Response channel signals.

Table B-25 Write Response channel signals

Signal	Type	Description	Connection information
BREADY_M	Output	Write response ready.	Connect to corresponding slave device, if populated.
BVALID_M	Input	Write response valid.	Connect to corresponding slave device, if populated, otherwise tie LOW.
BID_M[x:0] ^h	Input	Write response ID.	
BRESP_M[1:0]	Input	Write response.	
BUSER_M[3:0]	Input	User signal	
BTRACE_M	Input	-	
BIDUNQ_M	Input	Unique ID indicator signal.	Connect to corresponding slave device, if populated, otherwise tie LOW.
BCOMP_M	Input	Write/CMO observable. This signal is only present on SBSX. It is not present on HN-I.	Connect to corresponding slave device, if populated, otherwise tie LOW.
BPERSIST_M	Input	Data has been updated in persistent memory. This signal is only present on SBSX. It is not present on HN-I.	Connect to corresponding slave device, if populated, otherwise tie LOW.

The following table shows the Read Address channel signals.

^m The value of d = (((n + 1) / 8) - 1).

Table B-26 Read Address channel signals

Signal	Type	Description	Connection information
ARREADY_M	Input	Read address ready.	Connect to corresponding slave device, if populated, otherwise tie LOW.
ARVALID_M	Output	Read address valid.	Connect to corresponding slave device, if populated.
ARID_M[x:0]^h	Output	Read address ID.	
ARADDR_M[n:0]ⁱ	Output	Read address.	
ARLEN_M[7:0]	Output	Read burst length.	
ARSIZE_M[2:0]	Output	Read burst size.	
ARBURST_M[1:0]	Output	Read burst type.	
ARLOCK_M	Output	Read lock type.	
ARCACHE_M[3:0]	Output	Read cache type.	
ARUSER_M[n:0]	Output, where n = (REQ_RSVDC_WIDTH +METADATA_WIDTH (if enabled) + 2 (for DownStreamCache) - 1).	User signal.	
ARPROT_M[2:0]	Output	Read protection type.	
ARQOS_M[3:0]	Output	Read QoS value.	
ARSNOOP_M[3:0]	Output	Shareable read transaction type.	
ARDOMAIN_M[1:0]	Output	Read Shareability domain.	
ARTRACE_M	Output	-	
ARMPAM_M[10:0]	Output	MPAM signal. ARMPAM[0] MPAM_NS Security indicator, default = ARPROT[1] . ARMPAM[9:1] PARTID Partition identifier, default = 0x000. ARMPAM[10] PMG Performance monitor group, default = 0b0.	Connect to corresponding slave device, if populated.
ARIDUNQ_M	Output	Unique ID indicator signal.	Connect to corresponding slave device, if populated.
ARNSAID_M[3:0]	Output	Non-secure Access Identifier signal.	Connect to corresponding slave device, if populated.

The following table shows the Read Data channel signals.

Table B-27 Read Data channel signals

Signal	Type	Description	Connection information
RREADY_M	Output	Read data ready.	Connect to corresponding slave device, if populated.
RVALID_M	Input	Read data valid.	Connect to corresponding slave device, if populated, otherwise tie LOW.
RID_M[x:0]^b	Input	Read data ID.	Connect to corresponding slave device, if populated, otherwise tie LOW.
RDATA_M[127:0]/[255:0]	Input	Read data.	Connect to corresponding slave device, if populated, otherwise tie LOW.
RRESP_M[1:0]	Input	Read data response.	Connect to corresponding slave device, if populated, otherwise tie LOW.
RLAST_M	Input	Read data last transfer indication.	Connect to corresponding slave device, if populated, otherwise tie LOW.
RUSER_M[u:0]^k	Input	RUSER_M[0] is RDATACHK valid signal. If META_DATA_EN = 1, tie RUSER_M[u:1] to 0. RUSER_M[1] is Dirty indication. Metadata_width + 1. Metadata_width + 2 if metadata is enabled	Connect to corresponding slave device, if populated, otherwise tie LOW.
RPOISON_M[p:0]^l	Input	Poison signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
RDATACHK_M[d:0]^m	Input	Data check signal.	Connect to corresponding master device, if populated.
RTRACE_M	Input	Trace signal.	Connect to corresponding master device, if populated, otherwise tie LOW.
RIDUNQ_M	Input	Unique ID indicator signal.	Connect to corresponding master device, if populated, otherwise tie LOW.

————— Note ————

RUSER_M[0] acts as an **RDATACHK** valid signal when **DATACHECK_EN** parameter is enabled.

- If **RUSER_M[0]**=0, the SBSX or HN-I synthesizes the correct **RDATACHK** value before sending it on CHI read data response.
- If **RUSER_M[0]**=1, the SBSX or HN-I uses the **RDATACHK** pin value to drive CHI read data response.

If the **DATACHECK_EN** parameter is disabled, the **RUSER_M[0]** input is ignored.

————— Note ————

WUSER_M[0] acts as a **WDATACHK** valid signal. Since the SBSX or HN-I always drives the **WDATACHK** value, **WUSER_M[0]** is set to 1 when **DATACHECK_EN** parameter is enabled.

If the **DATACHECK_EN** parameter is not enabled, the **WUSER_M[0]** output is driven to 0.

B.4.3 Calculating the SBSX AxID signal widths

The width of the AxID signals in SBSX is 24 bits. However, the number of useful bits can be calculated using the equations provided for tracker optimization purposes.

Use the following equations to calculate the AWID width.

If SBSX_CMO_ON_AW = 1 (CMO_ON_WRITE is enabled):

- AWID = (2part_pcmo_coloring + PGroupID + log₂(NUM(RNF+RNI+RND+CXHA+HNF)) + log₂(ReqTracker_size))
- For example, the size of AWID based on the maximum values for each of the preceding parameters = 1 + 1 + 1 + 11 + 5 + 5 = 24.

AWID encoding: {Reserved[0:0], Reserved[0:0], 2-part_PCMO[0:0], ReturnNID[log₂(NUM(RNF+RNI+RND+CXHA+HNF))-1:0], PGroupID[4:0], RequestTrackerID[log₂(SBSX_NUM_REQS)-1:0]}

ARID encoding: {DartID[log₂(SBSX_NUM_DART)-1:0]}

————— Note —————

2part_pcmo_coloring indicates if the PCMO is a single part or two part PCMO on CHI.

Otherwise:

- AWID = log₂(ReqTracker_size)
- For example, the size of AWID based on the maximum value for the preceding parameter = 5.

The following equation indicates the maximum number of usable bits for ARID:

- MTU_color bit = ARIID[MSB]
- ARID (usable bits) = log₂(NUM_DART)

AWID encoding:

{Reserved[0:0], MTU_Color[0:0], 2-part_PCMO[0:0], ReturnNID[log₂(NUM(RNF+RNI+RND+CXHA+HNF))-1:0], GroupIDExt[2:0], PGroupID[4:0], RequestTrackerID[log₂(SBSX_NUM_REQS_PARAM)-1:0]}

ARID encoding:

MTU_color bit = ARIID[MSB].

{ DartID[log₂(SBSX_NUM_DART_PARAM)-1:0]}

B.4.4

HN-I and HN-P AxID signal properties and encodings

The size and encodings of the **AxID** signals on the AXI or ACE-Lite interface are different for HN-I and HN-P.

The term *peer-to-peer* refers to requests from an RN-I or RN-D that is connected to a PCIe-RC to an HN-I that is connected to a PCIe-RC or an HN-P. The pcie_mstr_present bit in the por_mn{i,d}_cfg_ctl register indicates whether an RN-I or RN-D is connected to a PCIe-RC. The physical_mem_en attribute can be enabled in SAM for address range where Normal Memory ordering is required. For example, Prefetchable MMIO/Non-prefetchable MMIO space which is mapped as 'Normal' memory

The following table shows the **AxID** properties and encodings for the HN-I and HN-P.

Table B-28 HN-I and HN-P AxID signal encodings

Device type	AxID width	Downstream memory type	Request source	AW or AR	Encoding	
HN-I	11	physical_mem_en = 0	Any	AWID or ARID	AxID[3:0]	HN-I SAM order region encoding.
					AxID[4]	Reserved.
HN-P	20	physical_mem_en = 0	Non peer-to-peer	AWID or ARID	AxID[5]	PCIe write coloring.
					AxID[7:6]	HN-I SAM address region encoding.
		physical_mem_en = 1	Non peer-to-peer	AWID or ARID	AxID[10:8]	Reserved.
					AxID[4:0]	UniqId[4:0].
HN-P	20	physical_mem_en = 0	Non peer-to-peer	AWID or ARID	AxID[5]	PCIe write coloring.
					AxID[7:6]	Address region encoding.
		physical_mem_en = 1	Non peer-to-peer	AWID or ARID	AxID[10:8]	UniqId[7:5].
					AxID[10:0]	Same as HN-I Device memory.
HN-P	20	physical_mem_en = 0	Non peer-to-peer	AWID or ARID	AxID[19:11]	Reserved.
					AxID[10:0]	Same as HN-I Normal memory.
		physical_mem_en = 1	Non peer-to-peer	AWID or ARID	AxID[19:11]	Reserved.
					AxID[10:0]	Same as HN-I Device memory.
HN-P	20	N/A (see Note)	Peer-to-peer	AWID	AWID[3:0]	Hash of PCIe RN-I AWID.
					AWID[5:4]	RN-I port number.
		N/A (see Note)	Peer-to-peer	AWID	AWID[11:6]	RN-I Logical ID.
					AWID[15:12]	CXHA Logical ID.
HN-P	20	N/A (see Note)	Peer-to-peer	AWID	AWID[16]	Remote RN-I.
					AWID[18:17]	Reserved.
		N/A (see Note)	Peer-to-peer	ARID	AWID[19]	Peer-to-peer coloring.
					ARID[HNP_RD_NUM_AXI_REQS_PARAM_LOG2-1:0]	HN-P read RDT entry.
HN-P	20	N/A (see Note)	Peer-to-peer	ARID	ARID[18:HNP_RD_NUM_AXI_REQS_PARAM_LOG2]	Reserved.
					AWID[19]	Peer-to-peer coloring.

————— Note —————

Peer-to-Peer traffic in HN-P does not generate AXIDs or order transactions based on downstream memory type.

B.4.5 A4S signals

The A4S interface signals are listed in the following tables.

Table B-29 A4S Transmit signals

Signal	Type	Description	Connection information
TXA4STREADY	Input	TXA4STREADY indicates that the slave can accept a transfer in the current cycle.	Connect from RXA4STREADY of the A4S slave, if populated, otherwise tie LOW.
TXA4STVALID	Output	TXA4STVALID indicates that the master is driving a valid transfer. A transfer takes place when both TXA4STVALID and TXA4STREADY are asserted.	Connect to RXA4STVALID of the A4S slave, if populated.
TXA4STDEST[7:0]	Output	0b00000000	TXA4STDEST is always zero.
TXA4STID[7:0]	Output	TXA4STID is the data stream identifier that indicates different streams of data.	Connect to RXA4STID of the A4S slave, if populated.
TXA4STDATA[63:0]	Output	TXA4STDATA is the primary payload that is used to provide the data that is passing across the interface.	Connect to RXA4STDATA of the A4S slave, if populated.
TXA4STSTRB[7:0]	Output	TXA4STSTRB is the byte qualifier that indicates whether the content of the associated byte of TXA4STDATA is processed as a data byte or a position byte.	Connect to RXA4STSTRB of the A4S slave, if populated.
TXA4STKEEP[7:0]	Output	TXA4STKEEP is the byte qualifier that indicates whether the content of the associated byte of TDATA is processed as part of the data stream. Associated bytes that have the TKEEP byte qualifier deasserted are null bytes and can be removed from the data stream.	Connect to RXA4STKEEP of the A4S slave, if populated.
TXA4STLAST	Output	TXA4STLAST indicates the boundary of a packet.	Connect to RXA4STLAST of the A4S slave, if populated.

Table B-30 A4S Receive signals

Signal	Type	Description	Connection information
RXA4STREADY	Output	RXA4STREADY indicates that the slave can accept a transfer in the current cycle.	Connect to TXA4STREADY of the A4S master, if populated.
RXA4STVALID	Input	RXA4STVALID indicates that the master is driving a valid transfer. A transfer takes place when both RXA4STVALID and RXA4STREADY are asserted.	Connect from TXA4STVALID of the A4S master, if populated, otherwise tie LOW.
RXA4STDEST[7:0]	Input	RXA4STDEST provides routing information for the data stream.	Connect from TXA4STDEST of the A4S master, if populated, otherwise tie LOW.
RXA4STID[7:0]	Input	RXA4STID is the data stream identifier that indicates different streams of data.	Connect from TXA4STID of the A4S master, if populated, otherwise tie LOW.
RXA4STRI[7:0]	Input	RXA4STRI is the chip to chip routing information that indicates RA ID of the other chip.	Connect from TXA4STRI of the A4S master, if populated, otherwise tie LOW.
RXA4STRI[7:0]	Input	-	Tie LOW.
RXA4STDATA[63:0]	Input	RXA4STDATA is the primary payload that is used to provide the data that is passing across the interface.	Connect from TXA4STDATA of the A4S master, if populated, otherwise tie LOW.
RXA4STSTRB[7:0]	Input	RXA4STSTRB is the byte qualifier that indicates whether the content of the associated byte of RXA4STDATA is processed as a data byte or a position byte.	Connect from TXA4STSTRB of the A4S master, if populated, otherwise tie LOW.

Table B-30 A4S Receive signals (continued)

Signal	Type	Description	Connection information
RXA4STKEEP[7:0]	Input	RXA4STKEEP is the byte qualifier that indicates whether the content of the associated byte of TDATA is processed as part of the data stream. Associated bytes that have the TKEEP byte qualifier deasserted are null bytes and are removed from the data stream.	Connect from TXA4STKEEP of the A4S master, if populated, otherwise tie LOW.
RXA4STLAST	Input	RXA4STLAST indicates the boundary of a packet.	Connect from TXA4STLAST of the A4S master, if populated, otherwise tie LOW.

B.5 APB interface signals

HN-D nodes have an APB interface to support the connection of an external APB master device.

The following table shows the APB signals.

Table B-31 APB signals

Signal	Direction	Description	Connection information
PADDR[31:0]	Input	Address that is associated with the APB transaction.	Connect to corresponding ports on external APB master device.
PPROT[2:0]	Input	Protection type of the transaction.	
PSEL	Input	Indicates that the slave device is selected and that a data transfer is required.	
PENABLE	Input	Enable. Indicates the second and subsequent cycles of an APB transfer.	
PWRITE	Input	Indicates that the access is a write when HIGH. Indicates that the access is a read when LOW.	
PWDATA[31:0]	Input	Write data.	
PSTRB[3:0]	Input	Write strobes.	
PREADY	Output	Ready.	
PRDATA[31:0]	Output	Read data.	
PSLVERR	Output	Indicates a transfer failure.	

B.6 Device population signals

B.7 Debug, trace, and PMU interface signals

Signals that aid debugging are included in CMN-700.

The following table shows the debug, trace, and PMU interface signals.

Note

All signal names in this section are only a root name indicated as **RootName**. CMN-700 interfaces use **RootName** within a more fully specified signal name as follows:

CMN-700 interface signal name == **RootName_NID#**, where # represents the node ID corresponding to the specific interface.

Table B-32 Debug, trace, and PMU interface signals

Signal	Type	Description	Connection information
ATCLKEN_NID<x>	Input	ATB clock enable, where <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-
ATREADY_NID<x>	Input	ATB device ready: 0 not ready 1 ready <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-
AFVALID_NID<x>	Input	FIFO flush request, where <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-
ATDATA[31:0]_NID<x>	Output	ATB data bus, where <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-
ATVALID_NID<x>	Output	ATB valid data: 0 no valid data 1 valid data <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-
ATBYTES[1:0]_NID<x>	Output	CoreSight ATB device data size: 0b00 1 byte 0b01 2 bytes 0b10 3 bytes 0b11 4 bytes <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-

Table B-32 Debug, trace, and PMU interface signals (continued)

Signal	Type	Description	Connection information
AFREADY_NID<x>	Output	FIFO flush acknowledge: 0 FIFO flush not complete 1 FIFO flush complete <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-
ATID[6:0]_NID<x>	Output	ATB trace source identification, where <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	-
DBGWATCHTRIGREQ_NID<x>	Output	Trigger output from DEM indicating assertion of a DT event. DBGWATCHTRIGREQ is asynchronous-safe, and operates in a four-phase handshake with DBGWATCHTRIGACK . <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	Connect to external debug and trace control logic.
DBGWATCHTRIGACK_NID<x>	Input	External acknowledgment of receipt of DBGWATCHTRIGREQ . DBGWATCHTRIGACK must be asynchronous-safe, and operates in a four-phase handshake with DBGWATCHTRIGREQ . <x> is the NodeID number for that HN-D/DTC or HN-T/DTC.	Connect to external debug and trace control logic, or tie LOW if DBGWATCHTRIGREQ is unused.
PMUSAPSHOTREQ	Input	External request that the live PMU counters are snapshot to the shadow registers. PMUSAPSHOTREQ must be asynchronous-safe, and operates in a four-phase handshake with PMUSAPSHOTACK .	Connect to external debug and trace control logic, or tie LOW if unused.
PMUSAPSHOTACK	Output	Indication that all live PMU counters have been copied to shadow registers and the contents can be read. PMUSAPSHOTACK is asynchronous-safe, and operates in a four-phase handshake with PMUSAPSHOTREQ .	Connect to external debug and trace control logic.
NIDEN	Input	Global enable for all debug, trace, and PMU functionality: 0 Disabled. 1 Enabled.	Tie or drive as appropriate to meet system security requirements.
SPNIDEN	Input	Global enable for Secure debug, trace, and PMU capability. Only applicable when NIDEN is enabled. 0 Disabled. 1 Enabled.	
TSVALUEB[63:0]	Input	Global system timestamp value in binary format.	Connect to external system timestamp counter output.

B.8 DFT and MBIST interface signals

Signals that support DFT and MBIST capabilities are included in CMN-700.

The following table shows the DFT signals.

Table B-33 DFT signals

Signal	Type	Description	Connection information
DFTCLKBYPASS	Input	Select the SLC RAM clock to follow the CMN-700 input clock, as applicable for each clock region.	Tie LOW if unused.
DFTCLKDISABLE[3:0]	Input	Disable clock regions during scan shift.	
DFTRAMHOLD	Input	Disable the RAM chip select during scan shift.	
DFTMCPHOLD	Input	Assert to prevent HN-F multicycle RAMs from clocking during capture cycles.	
DFTRSTDISABLE	Input	Disable internal synchronized reset during scan shift.	
DFTCGEN	Input	Scan shift enable. Forces on the clock grids during scan shift.	
DFTSCANMODE	Input	<p>During functional mode, the HN-F SLC and SF RAM set address and write data inputs satisfy RAM hold timing constraints using pipeline behavior. The set address and write data are only clocked and enabled the cycle before the RAMs are accessed, and are held the cycle that the RAM clock asserts.</p> <p>The RAM hold constraints are not guaranteed during ATPG test, because random data is shifted into the flops that control the set address and write data flop enables. This allows the set address and write data to change in the same cycle as a RAM access, violating the RAM hold constraints.</p> <p>This signal addresses the hold constraints during ATPG test. It is used to force the RAM set address and write data flop enables LOW in the cycle that RAM clocks are enabled during ATPG test.</p> <p>The combination of the functional pipeline behavior and this override logic enable holds MCPs to be used on the RAM set address and write data inputs in the implementation flow and during static timing analysis.</p>	

The following table shows the MBIST signals.

Table B-34 MBIST signals

Signal	Type	Description	Connection information
nMBISTRESET	Input	Primary reset to enter MBIST. Active-LOW. Must be HIGH during functional non-MBIST operation.	Tie HIGH if unused.
MBISTREQ	Input	SLC MBIST mode request.	Tie LOW if unused.

This section contains the following subsection:

- [B.8.1 Block-level ATPG signals on page Appx-B-1389](#).

B.8.1 Block-level ATPG signals

CMN-700 supports DFT using the ATPG methodology. The design contains various signals that are used to carry out ATPG testing.

The following table lists the ATPG signals at the HN-F por_hnf block-level.

Table B-35 HN-F block-level por_hnf ATPG signals

Signal	Direction	Description
DFTCLKBYPASS	Input	Bypasses stretch RAM clock for `STRETCH_L3RAMCLK configuration parameter. Functional mode = 0b0.
DFTRSTDISABLE	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
DFTCLKDISABLE	Input	Disables clock regions during test to save power. Functional mode = 0b0.
DFTCGEN	Input	Overrides clock gate shift. Functional mode = 0b0.
DFTRAMHOLD	Input	Blocks chip select to RAMs to preserve state. Functional mode = 0b0.
DFTMCPHOLD	Input	Limits number of multicycle path toggles during ATPG delay tests. Functional mode = 0b0.
DFTSCANMODE	Input	Prevents potential RAM input hold violations during ATPG. Functional mode = 0b0.
clk_por	Input	Functional clock.
nSKYRESET	Input	Functional reset. Active-LOW.
nMBISTRESET	Input	MBIST mode entry reset. Functional mode = 0b1. Active-LOW.

The following table lists the ATPG signals at the HN-D block-level (por_hnd), at the HN-I block-level (por_hni), and at the SBSX block-level (por_sbsx).

Table B-36 HN-D/HN-I/SBSX block-level por_hnd/por_hni/por_sbsx ATPG signals

Signal	Direction	Description
DFTRSTDISABLE	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
DFTCLKDISABLE	Input	Disables clock regions during test to save power. Functional mode = 0b0.
DFTCGEN	Input	Overrides clock gate shift. Functional mode = 0b0.
DFTCLKBYPASS	Input	Bypasses stretch RAM clock for `STRETCH_L3RAMCLK configuration parameter. Functional mode = 0b0.
clk_por	Input	Functional clock.
nCHIRESET	Input	Functional reset. Active-LOW.

The following table lists the ATPG signals at the RN-I block-level (por_rni) and at the RN-D block-level (por_rnd).

Table B-37 RN-I/RN-D block-level por_rni/por_rnd ATPG signals

Signal	Direction	Description
DFTCLKBYPASS	Input	Bypasses stretch RAM clock for `STRETCH_L3RAMCLK configuration parameter. Functional mode = 0b0.
DFTRSTDISABLE	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
DFTCLKDISABLE	Input	Disables clock gate regions during test to save power. Functional mode = 0b0.
DFTCGEN	Input	Overrides clock gate shift. Functional mode = 0b0.
DFTRAMHOLD	Input	Blocks chip select to RAMs to preserve state. Functional mode = 0b0.
DFTMCPHOLD	Input	Limits number of multicycle path toggles during ATPG delay tests. Functional mode = 0b0.

Table B-37 RN-I/RN-D block-level por_rni/por_rnd ATPG signals (continued)

Signal	Direction	Description
DFTSCANMODE	Input	Prevents violation of the HN-F RAM input multicycle hold paths on data and address RAM inputs. Signal is only present on MTSX ATPG interface.
clk_por	Input	Functional clock.
nCHIRESET	Input	Functional reset. Active-LOW.
nMBISTRESET	Input	MBIST mode entry reset. Functional mode = 0b1. Active-LOW.

The following table lists the ATPG signals as internal pins for SMXP blocks named por_smxp_*. Arm recommends that you use the drivers of these internal pins for any DFT purposes during synthesis.

Table B-38 SMXP block-level por_smxp_* ATPG signals

Signal	Direction	Description
u_mxp_misc.mxp_dfrstdisable	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
u_mxp_misc.mxp_dftclkdisable	Input	Disables clock regions during test to save power. Functional mode = 0b0.
u_mxp_misc.mxp_dftcgen	Input	Overrides clock gate shift. Functional mode = 0b0.
clk_por	Input	Functional clock.
u_mxp_misc.mxp_nporreset	Input	Functional reset. Active-LOW.

The following table lists the ATPG signals as internal nets for MCS blocks named por_mcsx and por_mcsy. Arm recommends that you use the drivers of these internal pins for any DFT purposes during synthesis.

Table B-39 MCS block-level por_mcsx/por_mcsy ATPG signals

Signal	Direction	Description
mcs_dfrstdisable	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
mcs_dftcgen	Input	Overrides clock gate shift. Functional mode = 0b0.
clk_por	Input	Functional clock.
mcs_nporreset	Input	Functional reset. Active-LOW.

The following table lists the ATPG signals as internal nets for DCS blocks named por_dcs_*. Arm recommends that you use the drivers of these internal pins for any DFT purposes during synthesis.

Table B-40 DCS block-level por_dcs_* ATPG signals

Signal	Direction	Description
des_dfrstdisable	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
dcs_dftcgen	Input	Overrides clock gate shift. Functional mode = 0b0.
clk_por	Input	Functional clock.
dcs_nporreset	Input	Functional reset. Active-LOW.

The following table lists the ATPG signals as internal nets for CCS blocks named por_ccs_*. Arm recommends that you use the drivers of these internal pins for any DFT purposes during synthesis.

Table B-41 CCS block-level por_ccs_* ATPG signals

Signal	Direction	Description
ccs_dftrstdisable	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
ccs_dftcgen	Input	Overrides clock gate shift. Functional mode = 0b0.
clk_por	Input	Functional clock.
ccs_nporreset	Input	Functional reset. Active-LOW.

The following table lists the ATPG signals as internal nets for CAL blocks named por_cal{2,4}_*. Arm recommends that you use the drivers of these internal pins for any DFT purposes during synthesis.

Table B-42 CAL block-level por_cal{2,4}_* ATPG signals

Signal	Direction	Description
cal_dftrstdisable	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
cal_dftcgen	Input	Overrides clock gate shift. Functional mode = 0b0.
clk_por	Input	Functional clock.
cal_nporreset	Input	Functional reset. Active-LOW.

The following table lists the ATPG signals at the CXRH block-level named por_cxrh.

Table B-43 CXRH block-level por_cxrh ATPG signals

Signal	Direction	Description
DFTRSTDISABLE	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
DFTCLKDISABLE	Input	Disables clock regions during test to save power. Functional mode = 0b0.
DFTCGEN	Input	Overrides clock gate shift. Functional mode = 0b0.
DFTRAMHOLD	Input	Blocks chip select to RAMs to preserve state. Functional mode = 0b0.
DFTMCPHOLD	Input	Limits number of multicycle path toggles during ATPG delay tests. Functional mode = 0b0.
clk_por	Input	Functional clock.
nCHIRESET	Input	Functional reset. Active-LOW.
nMBISTRESET	Input	MBIST mode entry reset. Functional mode = 0b1. Active-LOW.

The following table lists the ATPG signals at the CXLA block-level named por_cxla.

Table B-44 CXLA block-level por_cxla ATPG signals

Signal	Direction	Description
DFTRSTDISABLE	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
DFTCLKDISABLE	Input	Disables clock regions during test to save power. Functional mode = 0b0.
DFTCGEN	Input	Overrides clock gate shift. Functional mode = 0b0.
DFTMCPHOLD	Input	Limits number of multicycle path toggles during ATPG delay tests. Functional mode = 0b0.

Table B-44 CXLA block-level por_cxla ATPG signals (continued)

Signal	Direction	Description
CLK_CGL, CLK_CXS	Input	Functional clocks.
nRESET_CGL, nRESET_CXS	Input	Functional resets. Active-LOW.

The following table lists the ATPG signals at the PDB block-level (pdb_rnf) and at the CDB block-level (cdb_rnf/cdb_snf).

Table B-45 PDB/CDB block-level pdb_rnf/cdb_rnf/cdb_snf ATPG signals

Signal	Direction	Description
DFTRSTDISABLE	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
DFTCGEN	Input	Overrides clock gate shift. Functional mode = 0b0.
CLK_DEV, CLK_ICN	Input	Functional clocks.
RESETN_DEV, RESETN_ICN	Input	Functional resets. Active-LOW.

The following table lists the ATPG signals at the ADB block-level (adb_*).

Table B-46 ADB block-level adb_* ATPG signals

Signal	Direction	Description
DFTRSTDISABLE	Input	Disables internal synchronized reset during scan shift. Functional mode = 0b0.
DFTCGEN	Input	Overrides clock gate shift. Functional mode = 0b0.
CLK_S, CLK_M	Input	Functional clocks.
RESETN_S, RESETN_M	Input	Functional resets. Active-LOW.

B.9 Clock management signals

The following table shows the clock management Q-Channel signals.

Table B-47 Clock management Q-Channel signals

Signal	Type	Description	Connection information
QACTIVE_CLKCTL	Output	Indication that CMN-700 is active, and that the <i>External Clock Controller</i> (ExtCC) must not make a request for CMN-700 to prepare to stop the clocks.	Connect to external clock controller.
QREQn_CLKCTL	Input	Request from the ExtCC for the CMN-700 to prepare to stop the clocks. Active-LOW.	Connect to external clock controller or tie HIGH if unused.
QACCEPTn_CLKCTL	Output	Positive acknowledgment after receiving QREQn assertion indicating that CMN-700 has completed preparation to stop the clocks and that the ExtCC can stop the clocks. Active-LOW.	Connect to external clock controller.
QDENY_CLKCTL	Output	Negative acknowledgment after receiving QREQn assertion indicating that CMN-700 has refused the request from the ExtCC to prepare to stop the clocks.	

Table B-48 Clock management signals

Signal	Type	Description	Connection information
QACTIVE_CGLCLKCTL	Output	Indication that the CGL side of CMN-700 is active, and that the <i>External Clock Controller</i> (ExtCC) must not make a request for CMN-700 and corresponding CXG device to prepare to stop the clock CLK_CGL .	OR with QACTIVE_CGLCLKCTL (CXLA) and connect to external clock controller.
QREQn_CGLCLKCTL	Input	Request from the ExtCC for the CMN-700 to prepare to stop the clock CLK_CGL . Active-LOW.	Connect to external clock controller or tie HIGH if unused.
QACCEPTn_CGLCLKCTL	Output	Positive acknowledgment after receiving QREQn assertion indicating that CMN-700 has completed preparation to stop the clock CLK_CGL and that the ExtCC can stop the clock CLK_CGL . Active-LOW.	Connect to external clock controller.
QDENY_CGLCLKCTL	Output	Negative acknowledgment after receiving QREQn assertion indicating that CMN-700 has refused the request from the ExtCC to prepare to stop the clock CLK_CGL .	

B.10 Power management signals

This section contains information on power management signals.

The following table shows the power management signals for the logic power domain.

Table B-49 Power management signals for logic power domain

Signal	Type	Description	Connection information
PREQ_LOGIC	Input	Indicates a request for a power state transition.	Connect to external power management controller or tie LOW if unused.
PSTATE_LOGIC[4:0]	Input	The power state to which a transition is requested. ⁿ	Connect to external power management controller or tie to 5'b01000 if unused.
PACCEPT_LOGIC	Output	Indicates acknowledgment of the power state transition and completion of the power state transition within CMN-700.	Connect to external power management controller.
PDENY_LOGIC	Output	Indicates denial of the power state transition.	
PACTIVE_LOGIC	Output	Hint that indicates activity across the CMN-700. When LOW, indicates the possibility of entering static retention or the OFF state.	

————— Note ————

If **PACTIVE_LOGIC** is asserted, the system cannot be powered down.

ⁿ If Multicycle Path (MCP), the MCP duration must be ≤8 cycles to the last flop to receive this signal. This constraint is a requirement for implementation.

B.11 Interrupt and event signals

The following table shows the interrupt and event signals.

————— Note —————

All signal names in this section are root names, which are specified as **RootName**. CMN-700 interfaces use **RootName** within a fully specified signal name as follows:

CMN-700 interface signal name == **RootName_NID#**, where # represents the node ID corresponding to the specific interface.

Table B-50 Interrupt and event signals

Signal	Type	Description	Connection information
INTREQPPU	Output	Power state transition complete.	Connect to external interrupt control logic or Generic Interrupt Controller.
INTREQPMU_NID<x>	Output	PMU count overflow interrupt. NID indicates node ID where <x> represents the NodeID number for the HN-D/DTC or HN-T/DTC.	
INTREQERRNS	Output	Non-secure error handling interrupt.	
INTREQERRS	Output	Secure error handling interrupt.	
INTREQFAULTNS	Output	Non-secure fault handling interrupt.	
INTREQFAULTS	Output	Secure fault handling interrupt.	
INTREQMPAMERRNS	Output	Non-secure <i>Memory Partitioning And Monitoring</i> (MPAM) fault handling interrupt.	
INTREQMPAMERRS	Output	Secure MPAM fault handling interrupt.	

B.12 Configuration input signals

The following table shows the configuration input signals.

All of these signals must be stable at least ten cycles before deassertion of reset. These signals must remain stable throughout the operation of CMN-700, until a following reset assertion or power down, if any.

Table B-51 General configuration input signals

Signal	Type	Description	Connection information
CFGM_PERIPHBASE[47:28]	Input	Base address [47:28] of the CMN-700 configuration register space.	Tie as required for system memory map.
GICD_DESTID[15:0]	Input	A4S Logical ID of GICD connection.	Tie as required for CML GICD communication.

B.13 Processor event interface signals

Signals that support processor event interface capabilities are included in CMN-700.

The following table shows the processor event interface signals.

— Note —

All signal names in this section are only a root name indicated as **RootName**. CMN-700 interfaces use **RootName** within a more fully specified signal name as follows:

CMN-700 interface signal name == **RootName_NID#**, where # represents the node ID corresponding to the specific interface.

Processor event interface signals are present at RN-F, RN-I, and RN-D node locations.

Table B-52 Processor event interface signals

Signal	Type	Description	Connection information
EVENTIREQ	Output	Event input request for processor wake up from WFE state. Remains asserted until EVENTIACK is asserted, and is not reasserted until EVENTIACK is LOW.	Connect to EVENTIREQ input of processor.
EVENTIACK	Input	Event input request acknowledge. Must not be asserted until EVENTIREQ is HIGH, and then must remain asserted until after EVENTIREQ goes LOW.	Connect to EVENTIACK output of processor, or tie to EVENTIREQ output of CMN-700 if unused.
EVENTOREQ	Input	Event output request for processor wake up, triggered by SEV instruction. Must only be asserted when EVENTOACK is LOW, and then must remain HIGH until after EVENTOACK goes HIGH.	Connect to EVENTOREQ output of processor, or tie LOW if unused.
EVENTOACK	Output	Event output request acknowledge. Is not asserted until EVENTOREQ is HIGH, and then remains asserted until after EVENTOREQ goes LOW.	Connect to EVENTOACK input of processor.

— Note —

1. Event handling logic external to CMN-700 must handle **EVENT_OUT** from CHI processor. If system integration wants to connect the **EVENT_OUT** to CMN-700 **EVENTOREQ** or **EVENTOACK**, it is the responsibility of the integrator to design the necessary logic to stitch **EVENT_OUT**, which is a multicycle pulse, to the four-phase handshake pair, accounting for the asynchronous domain crossing.
2. Event handling logic external to CMN-700 can drive **EVENT_IN** of CHI processor.

B.14 CGL interface signals

The following tables describe CMN-700 CGL interface signals.

CCIX Gateway Link (CGL) interface (Credited Micro-Architecture interface between RA, HA, and LA components)

The following tables contain *CCIX Gateway Link* (CGL) interface signals.

————— Note ————

All signal names in this section are only a root name indicated as **RootName**. CMN-700 interfaces use **RootName** within a more fully specified signal name as follows:

CMN-700 interface signal name == **RootName_NID#**, where # represents the node ID corresponding to the specific interface.

The following table contains Transmit Memory Request interface signals.

————— Note ————

Values for ‘n’ and ‘m’ transmit signals ***REQDATFLIT** and ***SNPFLIT** are configuration-dependent.

Table B-53 Transmit Memory Request interface signals

Signal	Type	Description	Connection information
TXCGLREQDATFLITPEND	Output	Transmit Memory Request Early Flit Valid hint.	Connect to RXCGLREQDATFLITPEND of the corresponding CXLA.
TXCGLREQDATFLITV	Output	Transmit Memory Request Flit Valid.	Connect to RXCGLREQDATFLITV of the corresponding CXLA.
TXCGLREQDATFLIT[n:0]	Output	Transmit Memory Request Flit.	Connect to RXCGLREQDATFLIT of the corresponding CXLA.
TXCGLREQDATLCRDV	Input	Transmit Memory Request channel link layer credit.	Connect to RXCGLREQDATLCRDV of the corresponding CXLA.

The following table contains Transmit Snoop Request interface signals.

Table B-54 Transmit Snoop Request interface signals

Signal	Type	Description	Connection information
TXCGLSNPFLITPEND	Output	Transmit Snoop Request Early Flit Valid hint.	Connect to RXCGLSNPFLITPEND of the corresponding CXLA.
TXCGLSNPFLITV	Output	Transmit Snoop Request Flit Valid.	Connect to RXCGLSNPFLITV of the corresponding CXLA.
TXCGLSNPFLIT[m:0]	Output	Transmit Snoop Request Flit.	Connect to RXCGLSNPFLIT of the corresponding CXLA.
TXCGLSNPLCRDV	Input	Transmit Snoop Request channel link layer credit.	Connect to RXCGLSNPLCRDV of the corresponding CXLA.

The following table contains Transmit Memory Response interface signals.

Table B-55 Transmit Memory Response interface signals

Signal	Type	Description	Connection information
TXCGLREQRSPFLITPEND	Output	Transmit Memory Response Early Flit Valid hint.	Connect to RXCGLREQRSPFLITPEND of the corresponding CXLA.
TXCGLREQRSPFLITV	Output	Transmit Memory Response Flit Valid.	Connect to RXCGLREQRSPFLITV of the corresponding CXLA.
TXCGLREQRSPFLIT[35:0]	Output	Transmit Memory Response Flit.	Connect to RXCGLREQRSPFLIT of the corresponding CXLA.
TXCGLREQRSPLCRDV	Input	Transmit Memory Response channel link layer credit.	Connect to RXCGLREQRSPLCRDV of the corresponding CXLA.

The following table contains Transmit Snoop Response interface signals.

Table B-56 Transmit Snoop Response interface signals

Signal	Type	Description	Connection information
TXCGLSNPRSPFLITPEND	Output	Transmit Snoop Response Early Flit Valid hint.	Connect to RXCGLSNPRSPFLITPEND of the corresponding CXLA.
TXCGLSNPRSPFLITV	Output	Transmit Snoop Response Flit Valid.	Connect to RXCGLSNPRSPFLITV of the corresponding CXLA.
TXCGLSNPRSPFLIT[35:0]	Output	Transmit Snoop Response Flit.	Connect to RXCGLSNPRSPFLIT of the corresponding CXLA.
TXCGLSNPRSPLCRDV	Input	Transmit Snoop Response channel link layer credit.	Connect to RXCGLSNPRSPLCRDV of the corresponding CXLA.

The following table contains Transmit Snoop Data interface signals.

Table B-57 Transmit Snoop Data interface signals

Signal	Type	Description	Connection information
TXCGLSNPDATFLITPEND	Output	Transmit Snoop Data Early Flit Valid hint.	Connect to RXCGLSNPDATFLITPEND of the corresponding CXLA.
TXCGLSNPDATFLITV	Output	Transmit Snoop Data Flit Valid.	Connect to RXCGLSNPDATFLITV of the corresponding CXLA.
TXCGLSNPDATFLIT[559:0]	Output	Transmit Snoop Data Flit.	Connect to RXCGLSNPDATFLIT of the corresponding CXLA.
TXCGLSNPDATLCRDV	Input	Transmit Snoop Data channel link layer credit.	Connect to RXCGLSNPDATLCRDV of the corresponding CXLA.

The following table contains Transmit Memory Response interface signals.

Table B-58 Transmit Memory Response interface signals

Signal	Type	Description	Connection information
TXCGLRSPDATFLITPEND	Output	Transmit Memory Response Data Early Flit Valid hint.	Connect to RXCGLRSPDATFLITPEND of the corresponding CXLA.
TXCGLRSPDATFLITV	Output	Transmit Memory Response Data Flit Valid.	Connect to RXCGLRSPDATFLITV of the corresponding CXLA.

Table B-58 Transmit Memory Response interface signals (continued)

Signal	Type	Description	Connection information
TXCGLRSPDATFLIT[559:0]	Output	Transmit Memory Response Data Flit.	Connect to RXCGLRSPDATFLIT of the corresponding CXLA.
TXCGLRSPDATLCRDV	Input	Transmit Memory Response Data channel link layer credit.	Connect to RXCGLRSPDATLCRDV of the corresponding CXLA.

The following table contains Transmit Protocol Credit interface signals.

Table B-59 Transmit Protocol Credit interface signals

Signal	Type	Description	Connection information
TXCGLPCRDFLITPEND	Output	Transmit Protocol Credit Early Flit Valid hint.	Connect to RXCGLPCRDFLITPEND of the corresponding CXLA.
TXCGLPCRDFLITV	Output	Transmit Protocol Credit Flit Valid.	Connect to RXCGLPCRDFLITV of the corresponding CXLA.
TXCGLPCRDFLIT[50:0]	Output	Transmit Protocol Credit Flit.	Connect to RXCGLPCRDFLIT of the corresponding CXLA.
TXCGLPCRDLCRDV	Input	Transmit Protocol Credit channel link layer credit.	Connect to RXCGLPCRDLCRDV of the corresponding CXLA.

The following table contains Receive Memory Request interface signals.

————— Note —————

Values for ‘n’ and ‘m’ receive signals *REQDATFLIT and *SNPFLIT are configuration-dependent.

Table B-60 Receive Memory Request interface signals

Signal	Type	Description	Connection information
RXCGLREQDATFLITPEND	Input	Receive Memory Request Early Flit Valid hint.	Connect to TXCGLREQDATFLITPEND of the corresponding CXLA.
RXCGLREQDATFLITV	Input	Receive Memory Request Flit Valid.	Connect to TXCGLREQDATFLITV of the corresponding CXLA.
RXCGLREQDATFLIT[n:0]	Input	Receive Memory Request Flit.	Connect to TXCGLREQDATFLIT of the corresponding CXLA.
RXCGLREQDATLCRDV	Output	Receive Memory Request channel link layer credit.	Connect to TXCGLREQDATLCRDV of the corresponding CXLA.

The following table contains Receive Snoop Request interface signals.

Table B-61 Receive Snoop Request interface signals

Signal	Type	Description	Connection information
RXCGLSNPFLITPEND	Input	Receive Snoop Request Early Flit Valid hint.	Connect to TXCGLSNPFLITPEND of the corresponding CXLA.
RXCGLSNPFLITV	Input	Receive Snoop Request Flit Valid.	Connect to TXCGLSNPFLITV of the corresponding CXLA.

Table B-61 Receive Snoop Request interface signals (continued)

Signal	Type	Description	Connection information
RXCGLSNPFLIT[m:0]	Input	Receive Snoop Request Flit.	Connect to TXCGLSNPFLIT of the corresponding CXLA.
RXCGLSNPLCRDV	Output	Receive Snoop Request channel link layer credit.	Connect to TXCGLSNPLCRDV of the corresponding CXLA.

The following table contains Receive Memory Response interface signals.

Table B-62 Receive Memory Response interface signals

Signal	Type	Description	Connection information
RXCGLREQRSRSPFLITPEND	Input	Receive Memory Response Early Flit Valid hint.	Connect to TXCGLREQRSRSPFLITPEND of the corresponding CXLA.
RXCGLREQRSRSPFLITV	Input	Receive Memory Response Flit Valid.	Connect to TXCGLREQRSRSPFLITV of the corresponding CXLA.
RXCGLREQRSRSPFLIT[35:0]	Input	Receive Memory Response Flit.	Connect to TXCGLREQRSRSPFLIT of the corresponding CXLA.
RXCGLREQRSRSPPLCRDV	Output	Receive Memory Response channel link layer credit.	Connect to TXCGLREQRSRSPPLCRDV of the corresponding CXLA.

The following table contains Receive Snoop Response interface signals.

Table B-63 Receive Snoop Response interface signals

Signal	Type	Description	Connection information
RXCGLSNPRSPFLITPEND	Input	Receive Snoop Response Early Flit Valid hint.	Connect to TXCGLSNPRSPFLITPEND of the corresponding CXLA.
RXCGLSNPRSPFLITV	Input	Receive Snoop Response Flit Valid.	Connect to TXCGLSNPRSPFLITV of the corresponding CXLA.
RXCGLSNPRSPFLIT[35:0]	Input	Receive Snoop Response Flit.	Connect to TXCGLSNPRSPFLIT of the corresponding CXLA.
RXCGLSNPRSPPLCRDV	Output	Receive Snoop Response channel link layer credit.	Connect to TXCGLSNPRSPPLCRDV of the corresponding CXLA.

The following table contains Receive Snoop Data interface signals.

Table B-64 Receive Snoop Data interface signals

Signal	Type	Description	Connection information
RXCGLSNPDATFLITPEND	Input	Receive Snoop Data Early Flit Valid hint.	Connect to TXCGLSNPDATFLITPEND of the corresponding CXLA.
RXCGLSNPDATFLITV	Input	Receive Snoop Data Flit Valid.	Connect to TXCGLSNPDATFLITV of the corresponding CXLA.

Table B-64 Receive Snoop Data interface signals (continued)

Signal	Type	Description	Connection information
RXCGLSNPDATFLIT[559:0]	Input	Receive Snoop Data Flit.	Connect to TXCGLSNPDATFLIT of the corresponding CXLA.
RXCGLSNPDATLCRDV	Output	Receive Snoop Data channel link layer credit.	Connect to TXCGLSNPDATLCRDV of the corresponding CXLA.

The following table contains Receive Memory Response interface signals.

Table B-65 Receive Memory Response interface signals

Signal	Type	Description	Connection information
RXCGLRSPDATFLITPEND	Input	Receive Memory Response Data Early Flit Valid hint.	Connect to TXCGLRSPDATFLITPEND of the corresponding CXLA.
RXCGLRSPDATFLITV	Input	Receive Memory Response Data Flit Valid.	Connect to TXCGLRSPDATFLITV of the corresponding CXLA.
RXCGLRSPDATFLIT[559:0]	Input	Receive Memory Response Data Flit.	Connect to TXCGLRSPDATFLIT of the corresponding CXLA.
RXCGLRSPDATLCRDV	Output	Receive Memory Response Data channel link layer credit.	Connect to TXCGLRSPDATLCRDV of the corresponding CXLA.

The following table contains Receive Protocol Credit interface signals.

Table B-66 Receive Protocol Credit interface signals

Signal	Type	Description	Connection information
RXCGLPCRDFLITPEND	Input	Receive Protocol Credit Early Flit Valid hint.	Connect to TXCGLPCRDFLITPEND of the corresponding CXLA.
RXCGLPCRDFLITV	Input	Receive Protocol Credit Flit Valid.	Connect to TXCGLPCRDFLITV of the corresponding CXLA.
RXCGLPCRDFLIT[50:0]	Input	Receive Protocol Credit Flit.	Connect to TXCGLPCRDFLIT of the corresponding CXLA.
RXCGLPCRDLCRDV	Output	Receive Protocol Credit channel link layer credit.	Connect to TXCGLPCRDLCRDV of the corresponding CXLA.

The following table contains Receive and Transmit channel LinkActive interface signals.

Table B-67 Receive and Transmit channel LinkActive interface signals

Signal	Type	Description	Connection information
RXCGLLINKACTIVEREQ	Input	Receive channel LinkActive request from CXLA.	Connect to TXCGLLINKACTIVEREQ of the corresponding CXLA.
RXCGLLINKACTIVEACK	Output	Receive channel LinkActive acknowledgement to CXLA.	Connect to TXCGLLINKACTIVEACK of the corresponding CXLA.

Table B-67 Receive and Transmit channel LinkActive interface signals (continued)

Signal	Type	Description	Connection information
TXCGLLINKACTIVEREQ	Output	Transmit channel LinkActive request to CXLA.	Connect to RXCGLLINKACTIVEREQ of the corresponding CXLA.
TXCGLLINKACTIVEACK	Input	Transmit channel LinkActive acknowledgement from CXLA.	Connect to RXCGLLINKACTIVEACK of the corresponding CXLA.

The following table contains RXCGLSACTIVE and TXCGLSACTIVE interface signals.

Table B-68 RXCGLSACTIVE and TXCGLSACTIVE interface signals

Signal	Type	Description	Connection information
RXCGLSACTIVE	Input	Indication from CXLA that it has one or more outstanding protocol-layer transactions. RXSACTIVE must remain asserted throughout the lifetime of the transaction.	Connect to TXCGLSACTIVE of the corresponding CXLA.
TXCGLSACTIVE	Output	Indication to CXLA that CXRH has one or more outstanding protocol-layer transactions. TXSACTIVE remains asserted throughout the lifetime of the transaction.	Connect to RXCGLSACTIVE of the corresponding CXLA.

B.15 CXLA configuration interface signals

Signals that support CXLA configuration are included in CMN-700.

The following table shows the CXLA configuration interface signals.

— Note —

All signal names in this section are only a root name indicated as **RootName**. CMN-700 interfaces use **RootName** within a more fully specified signal name as follows:

CMN-700 interface signal name == **RootName_NID#**, where # represents the node ID corresponding to the specific interface.

— Note —

For the top-level CXLA interface signals, see the *Configuration and Integration Manual*, which is only available to licensees.

Table B-69 CXLA configuration interface signals

Signal	Type	Description	Connection information
RXPUBFLITPEND	Input	Receive channel early flit valid hint	Connect to TXPUBFLITPEND of the corresponding CXLA device, if populated. Otherwise, tie LOW.
RXPUBFLITV	Input	Receive channel flit valid	Connect to TXPUBFLITV of the corresponding CXLA device, if populated. Otherwise, tie LOW.
RXPUBFLIT[34:0]	Input	Receive channel flit	Connect to TXPUBFLIT[n:0] of the corresponding CXLA device, if populated. Otherwise, tie LOW.
RXPUBLINKFLIT	Input	Receive channel link flit	Connect to TXPUBLINKFLIT of the corresponding CXLA device, if populated. Otherwise, tie LOW.
RXPUBLCRDV_RP1	Output	Receive channel link layer credit	Connect to TXPUBLCRDV_RP1 of the corresponding CXLA device, if populated.
RXPUBLINKACTIVEREQ	Input	Receive channel LinkActive request	Connect to TXPUBLINKACTIVEREQ of the corresponding CXLA device, if populated. Otherwise, tie LOW.
RXPUBLINKACTIVEACK	Output	Receive channel LinkActive acknowledge	Connect to TXPUBLINKACTIVEACK of the corresponding CXLA device, if populated.
TXPUBFLITPEND	Output	Transmit channel flit valid	Connect to RXPUBFLITPEND of the corresponding CXLA device, if populated.
TXPUBFLITV	Output	Transmit channel early flit valid hint	Connect to RXPUBFLITV of the corresponding CXLA device, if populated.
TXPUBFLIT[34:0]	Output	Transmit channel flit	Connect to RXPUBFLIT[34:0] of the corresponding CXLA device, if populated.
TXPUBLINKFLIT	Output	Transmit channel link flit	Connect to TXPUBLINKFLIT of the corresponding CXLA device, if populated.
TXPUBLCRDV_RP1	Input	Transmit channel link layer credit	Connect to TXPUBLCRDV_RP1 of the corresponding CXLA device, if populated, otherwise tie LOW.
TXPUBLINKACTIVEREQ	Output	Transmit channel LinkActive request	Connect to TXPUBLINKACTIVEREQ of the corresponding CXLA device, if populated.

Table B-69 CXLA configuration interface signals (continued)

Signal	Type	Description	Connection information
TXPUBLINKACTIVEACK	Input	Transmit channel LinkActive acknowledge	Connect to RXPUBLINKACTIVEACK of the corresponding CXLA device, if populated, otherwise tie LOW.
TXPUBCFGACTIVE	Output	Transmit channel Configuration Active	Connect to RXPUBCFGACTIVE of the corresponding CXLA device, if populated.

Appendix C

Revisions

This appendix describes the technical changes between released issues of this manual.

It contains the following section:

- [*C.1 Revisions* on page Appx-C-1408.](#)

C.1 Revisions

This section lists the differences between released versions of the document.

Table C-1 Issue 0000-01

Change	Location
First release	-