

Rockchip
RK1808
Technical Reference Manual
Part2

Revision 1.0
Jan. 2019

Rockchip Confidential

Revision History

| Date | Revision | Description |
|-------------|-----------------|--------------------|
| 2019-01-21 | 1.0 | Initial Release |
| | | |
| | | |
| | | |
| | | |

Rockchip Confidential

Table of Content

| | |
|--|-----|
| Table of Content | 3 |
| Figure Index | 5 |
| Table Index..... | 7 |
| Warranty Disclaimer..... | 8 |
| Chapter 1 Interconnect..... | 9 |
| 1.1 Overview | 9 |
| 1.2 Block Diagram | 9 |
| 1.3 Function Description | 9 |
| 1.4 Register Description..... | 13 |
| 1.5 Application Notes | 29 |
| Chapter 2 Dynamic Memory Interface (DMC) | 31 |
| 2.1 Overview | 31 |
| 2.2 Block Diagram | 32 |
| 2.3 Function Description | 32 |
| 2.4 Register Description..... | 33 |
| 2.5 Interface Description | 167 |
| 2.6 Application Notes | 168 |
| Chapter 3 Encryption and Decryption(Crypto) | 174 |
| 3.1 Overview | 174 |
| 3.2 Block Diagram | 174 |
| 3.3 Register description..... | 175 |
| 3.4 Application Note..... | 204 |
| Chapter 4 Multi-format Video Encoder and Decoder | 208 |
| 4.1 Overview | 208 |
| 4.2 Block Diagram | 208 |
| 4.3 Video frame format | 209 |
| 4.4 Function Description | 212 |
| 4.5 Register Description | 216 |
| 4.6 Application Notes | 314 |
| Chapter 5 Video Output Processor (VOP_LITE) | 315 |
| 5.1 Overview | 315 |
| 5.2 Block Diagram | 316 |
| 5.3 Function Description | 316 |
| 5.4 Register Description..... | 321 |
| 5.5 Timing Diagram | 345 |
| 5.6 Interface Description | 346 |
| 5.7 Application Notes | 347 |
| Chapter 6 Video Output Processor (VOP_RAW)..... | 350 |
| 6.1 Overview | 350 |
| 6.2 Function Description | 351 |
| 6.3 Timing Diagram | 353 |
| 6.4 Register Description | 354 |
| Chapter 7 Raster Graphic Acceleration(RGA)..... | 373 |
| 7.1 Overview | 373 |
| 7.2 Block Diagram | 374 |
| 7.3 Function Description | 375 |
| 7.4 Register Description | 379 |
| 7.5 Application Notes | 403 |
| Chapter 8 USB2.0 Host Controller | 406 |
| 8.1 Overview | 406 |
| 8.2 Block Diagram | 406 |
| 8.3 Function Description | 406 |
| 8.4 Register Description..... | 407 |

| | |
|---|-----|
| 8.5 Interface Description | 407 |
| 8.6 Application Notes | 407 |
| Chapter 9 USB3.0 Controller | 408 |
| 9.1 Overview | 408 |
| 9.2 Block Diagram | 409 |
| 9.3 Function Description | 409 |
| 9.4 Register Description | 410 |
| 9.5 Interface Description | 477 |
| 9.6 Application Notes | 478 |
| Chapter 10 PCIe Controller..... | 479 |
| 10.1 Overview..... | 479 |
| 10.2 Block Diagram | 479 |
| 10.3 Function Description | 480 |
| 10.4 Interface Description..... | 482 |
| 10.5 Register Description..... | 483 |
| 10.6 Application Notes..... | 793 |
| 10.7 PCIe Appendix | 822 |
| Chapter 11 Voice Activity Detect (VAD)..... | 824 |
| 11.1 Overview..... | 824 |
| 11.2 Block Diagram | 824 |
| 11.3 Function Description | 825 |
| 11.4 Register Description..... | 827 |
| 11.5 Application Notes..... | 835 |
| Chapter 12 MIPI DSI Host Controller | 837 |
| 12.1 Overview..... | 837 |
| 12.2 Block Diagram | 837 |
| 12.3 Function Description | 838 |
| 12.4 Register Description..... | 845 |
| 12.5 Application Notes..... | 878 |
| Chapter 13 MIPI CSI Transmitter Device(CSI2TX)..... | 881 |
| 13.1 Overview..... | 881 |
| 13.2 Block Diagram | 881 |
| 13.3 Function Description | 882 |
| 13.4 Register Description..... | 882 |
| 13.5 Interface Timing..... | 906 |
| 13.6 Application Notes..... | 908 |
| Chapter 14 MIPI CSI-2 Host Controller..... | 909 |
| 14.1 Overview..... | 909 |
| 14.2 Block Diagram | 909 |
| 14.3 Function Description | 909 |
| 14.4 Register Description..... | 911 |
| 14.5 Application Notes..... | 923 |
| Chapter 15 MIPI DPHY TX | 924 |
| 15.1 Overview..... | 924 |
| 15.2 Block Diagram | 924 |
| 15.3 Function Description | 925 |
| 15.4 Register Description..... | 926 |
| 15.5 Application Notes..... | 933 |
| Chapter 16 MIPI DPHY RX | 934 |
| 16.1 Overview..... | 934 |
| 16.2 Block Diagram | 934 |
| 16.3 Register Description..... | 935 |
| 16.4 Interface Description..... | 946 |
| 16.5 Application Notes..... | 947 |

Figure Index

| | |
|--|-----|
| Fig. 1-1 Interconnect diagram | 9 |
| Fig. 1-2 Memory Scheduler diagram | 10 |
| Fig. 1-3 Idle request | 30 |
| Fig. 3-1 Crypto Architecture | 174 |
| Fig. 3-2 LLI DMA Usage | 205 |
| Fig. 3-3 AES-HASH-RX mode..... | 207 |
| Fig. 3-4 AES-HASH-TX mode | 207 |
| Fig. 4-1 Decoder in SOC | 208 |
| Fig. 4-2 VPU Block Diagram | 209 |
| Fig. 4-3 VCODEC YCbCr 4:2:0 planar format | 209 |
| Fig. 4-4 VCODEC YCbCr 4:2:0 Semi-planar format | 210 |
| Fig. 4-5 VCODEC Tile scan mode..... | 210 |
| Fig. 4-6 VCODEC YCbCr4:2:2 Interleaved format | 211 |
| Fig. 4-7 VCODEC AYCbCr 4:4:4 Interleaved format..... | 211 |
| Fig. 4-8 VCODEC RGB 16bpp format..... | 212 |
| Fig. 4-9 Structure of two-level page table | 212 |
| Fig. 4-10 The dataflow of JPEG decoder | 214 |
| Fig. 4-11 Video Encoder Dataflow | 215 |
| Fig. 5-1 VOP Block Diagram | 316 |
| Fig. 5-2 VOP Frame Buffer Data Format | 316 |
| Fig. 5-3 VOP Win1 Palette (8bpp)..... | 316 |
| Fig. 5-4 VOP Internal DMA | 317 |
| Fig. 5-5 VOP Virtual Display Mode | 317 |
| Fig. 5-6 VOP Overlay Display..... | 318 |
| Fig. 5-7 VOP Alpha blending | 318 |
| Fig. 5-8 VOP BCSH Diagram | 319 |
| Fig. 5-9 VOP Gamma LUTs | 319 |
| Fig. 5-10 VOP Replication..... | 320 |
| Fig. 5-11 VOP dithering | 320 |
| Fig. 5-12 FRC pattern diagram..... | 321 |
| Fig. 5-13 VOP RGB interface timing setting..... | 345 |
| Fig. 5-14 VOP RGB Interface Timing(SDR)..... | 346 |
| Fig. 5-15 VOP RGB Mode Programming Flow..... | 348 |
| Fig. 6-1 VOP Block Diagram | 351 |
| Fig. 6-2 VOP Internal DMA | 351 |
| Fig. 6-3 VOP Virtual Display Mode | 352 |
| Fig. 6-4 VOP PDAF vblank mode | 352 |
| Fig. 6-5 VOP PDAF hvblank mode..... | 353 |
| Fig. 6-6 VOP PDAF interweave mode | 353 |
| Fig. 6-7 VOP camera interface timing setting | 353 |
| Fig. 6-8 VOP RGB Interface Timing(SDR) | 354 |
| Fig. 7-1 RGA Block Diagram | 374 |
| Fig. 7-2 RGA2 in SOC..... | 374 |
| Fig. 7-3 RGA Input Data Format | 375 |
| Fig. 7-4 RGA Dither effect..... | 376 |
| Fig. 7-5 Layer0 alpha blending calculate flow | 377 |
| Fig. 7-6 Layer1 alpha blending calculate flow | 378 |
| Fig. 7-7 RGA Gradient Fill | 378 |
| Fig. 7-8 RGAsoftware main register-region | 403 |
| Fig. 7-9 RGA command line and command counter | 404 |
| Fig. 7-10 RGA command sync generation | 404 |
| Fig. 7-11 The size constraint among A B C | 404 |
| Fig. 8-1 USB2.0 Host Controller Block Diagram | 406 |
| Fig. 9-1 USB3.0 OTG Block Diagram | 409 |
| Fig. 10-1 PCIe Block Diagram | 480 |

| | |
|---|-----|
| Fig. 10-2 System Level View of PCIe EDMA | 481 |
| Fig. 10-3 PCIe Core Architecture Overview | 481 |
| Fig. 10-4 PCIe PHY Reference Clock Structure..... | 794 |
| Fig. 10-5 Delaying the PCIe Hot Reset | 796 |
| Fig. 10-6 PCIe Hot Reset Control Structure..... | 796 |
| Fig. 10-7 PCIe Hot Reset Control Structure..... | 799 |
| Fig. 10-8 Inbound BAR Match Mode..... | 803 |
| Fig. 10-9 System Level View of PCIe Embedded DMA..... | 806 |
| Fig. 10-10 Write Interrupt Generation - Non Linked List Mode | 807 |
| Fig. 10-11 Read Interrupt Generation - Non Linked List Mode..... | 808 |
| Fig. 10-12 Write Interrupt Generation - Linked List Mode..... | 809 |
| Fig. 10-13 Read Interrupt Generation - Linked List Mode | 810 |
| Fig. 10-14 Linked List Element/Descriptor Structure in Local Memory | 811 |
| Fig. 10-15 Write Transfer: SAR and DAR for Write Channel | 813 |
| Fig. 10-16 Read Transfer: SAR and DAR for Write Channel..... | 813 |
| Fig. 11-1 VAD Block Diagram | 825 |
| Fig. 11-2 VAD_det Block Diagram | 825 |
| Fig. 11-3 VAD usage flow..... | 835 |
| Fig. 12-1 MIPI DSI HOST Controller architecture | 838 |
| Fig. 12-2 24 bpp APB Pixel to Byte Organization..... | 841 |
| Fig. 12-3 18 bpp APB Pixel to Byte Organization..... | 841 |
| Fig. 12-4 16 bpp APB Pixel to Byte Organization..... | 841 |
| Fig. 12-5 12 bpp APB Pixel to Byte Organization..... | 842 |
| Fig. 12-6 8bpp APB Pixel to Byte Organization | 842 |
| Fig. 12-7 Command Transmission Periods within the Image Area | 842 |
| Fig. 12-8 Location in the Image Area | 844 |
| Fig. 13-1 CSI2TX Block Diagram | 882 |
| Fig. 13-2 DPCC pattern..... | 882 |
| Fig. 13-3 Camera Interface Timing without pdaf..... | 906 |
| Fig. 13-4 Camera Interface Timing with pdaf | 907 |
| Fig. 13-5 IDI Interface Typical Timing | 907 |
| Fig. 13-6 APB Interface Typical Read Timing | 907 |
| Fig. 13-7 APB Interface Typical Write Timing | 908 |
| Fig. 13-8 DPHY Interface Typical Write Timing | 908 |
| Fig. 14-1 MIPI CSI-2 Host Controller architecture | 909 |
| Fig. 15-1 MIPI DPHY TX detailed block diagram..... | 924 |
| Fig. 15-2 MIPI DPHY TX VOPLITE and VOPRAW connection | 925 |
| Fig. 16-1 MIPI DPHY RX Block Diagram..... | 934 |

Table Index

| | |
|--|-----|
| Table 1-1 Internal Priority Setting | 11 |
| Table 1-2 External Enable Register Address | 11 |
| Table 1-3 Probe | 12 |
| Table 2-1 DDR IO description | 167 |
| Table 2-2 DDR IO Mapping Table | 168 |
| Table 2-3 CK/CMD Driver output resistance | 170 |
| Table 2-4 DM, DQ Signal Drive Strength Register | 170 |
| Table 2-5 DDR3 1.5V DQ/DQS/CMD Driver and ODT resistance | 171 |
| Table 2-6 DDR/LPDDR2/3 1.2V DQ/DQS/CMD driver and ODT resistance | 171 |
| Table 2-7 Per-bit de-skew tuning resolution | 171 |
| Table 3-1 Crypto Clock & Reset Description | 204 |
| Table 3-2 Crypto Clock & Reset Description | 204 |
| Table 3-3 Crypto Performance Description | 204 |
| Table 3-4 LLI Item Description..... | 205 |
| Table 3-5 LLI Item dma_ctl Description..... | 205 |
| Table 3-6 LLI Item user_define Description..... | 206 |
| Table 3-7 LLI Item user_define Description..... | 206 |
| Table 3-8 LLI Item user_define Description..... | 206 |
| Table 4-1 H.264 feature..... | 212 |
| Table 4-2 JPEG features..... | 213 |
| Table 4-3 Video encoder H.264 feature | 214 |
| Table 4-4 JPGE features..... | 215 |
| Table 4-5 Base address of config..... | 216 |
| Table 4-6 Base address value | 216 |
| Table 5-1 VOP Data Swap of Win1 | 317 |
| Table 5-2 VOP Control Pins Definition..... | 346 |
| Table 5-3 Gather configuration for all format | 347 |
| Table 5-4 Effective immediately register table | 349 |
| Table 7-1 RGAROP Boolean operations | 379 |
| Table 8-1 USB2.0 Host Controller Address Mapping..... | 407 |
| Table 8-2 USB2.0 PHY Interface Description | 407 |
| Table 9-1 USB3 Address Mapping | 410 |
| Table 9-2 USB2.0 PHY and PCIe&SS PHY Interface Description | 477 |
| Table 10-1 PCIe Interface Description | 482 |
| Table 10-2 PCIe Client and Core Register Address Mapping | 483 |
| Table 10-3 PCIe Core Register Map in RC mode | 483 |
| Table 10-4 PCIe Core Register Map in EP mode | 483 |
| Table 10-5 PCIe PHY reference clock registers..... | 795 |
| Table 10-6 Determination of Match Mode | 802 |
| Table 10-7 Write DMA Transfer Example | 814 |
| Table 10-8 Read DMA Transfer Example | 815 |
| Table 10-9 Target L1 Substate as a Function of Software Controls (USP)..... | 820 |
| Table 10-10 Target L1 Substate as a Function of Software Controls (DSP)..... | 820 |
| Table 10-11 PCIe Interrupt Table..... | 821 |
| Table 12-1 Color table | 839 |
| Table 14-1 Supported Camera Settings | 910 |
| Table 14-2 Errors Identified by the CSI-2 Host Controller..... | 910 |
| Table 15-1 Function of grf bits in MIPI mode..... | 925 |
| Table 16-1 MIPI DPHY RX Interface Description | 946 |

Warranty Disclaimer

Rockchip Electronics Co., Ltd makes no warranty, representation or guarantee (expressed, implied, statutory, or otherwise) by or with respect to anything in this document, and shall not be liable for any implied warranties of non-infringement, merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

Information furnished is believed to be accurate and reliable. However, Rockchip Electronics Co., Ltd assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use.

Rockchip Electronics Co., Ltd's products are not designed, intended, or authorized for using as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Rockchip Electronics Co., Ltd's product could create a situation where personal injury or death may occur, should buyer purchase or use Rockchip Electronics Co., Ltd's products for any such unintended or unauthorized application, buyers shall indemnify and hold Rockchip Electronics Co., Ltd and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorized use, even if such claim alleges that Rockchip Electronics Co., Ltd was negligent regarding the design or manufacture of the part.

Copyright and Patent Right

Information in this document is provided solely to enable system and software implementers to use Rockchip Electronics Co., Ltd's products. There are no expressed or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Rockchip Electronics Co., Ltd does not convey any license under its patent rights nor the rights of others.

All copyright and patent rights referenced in this document belong to their respective owners and shall be subject to corresponding copyright and patent licensing requirements.

Trademarks

Rockchip and Rockchip™ logo and the name of Rockchip Electronics Co., Ltd's products are trademarks of Rockchip Electronics Co., Ltd. and are exclusively owned by Rockchip Electronics Co., Ltd. References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

Confidentiality

The information contained herein (including any attachments) is confidential. The recipient hereby acknowledges the confidentiality of this document, and except for the specific purpose, this document shall not be disclosed to any third party.

Reverse engineering or disassembly is prohibited.

ROCKCHIP ELECTRONICS CO., LTD. RESERVES THE RIGHT TO MAKE CHANGES IN ITS PRODUCTS OR PRODUCT SPECIFICATIONS WITH THE INTENT TO IMPROVE FUNCTION OR DESIGN AT ANY TIME AND WITHOUT NOTICE AND IS NOT REQUIRED TO UPDATE THIS DOCUMENTATION TO REFLECT SUCH CHANGES.

Copyright © 2019 Rockchip Electronics Co., Ltd.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electric or mechanical, by photocopying, recording, or otherwise, without the prior written consent of Rockchip Electronics Co., Ltd.

Interconnect

Chapter 1 Interconnect

1.1 Overview

The chip-level interconnect enables communication among the modules and subsystems in the device. It supports the following features:

- Cross-bar exchange network
- Embedded memory scheduler for DDR transaction reorder to maximum DDR efficiency
- Priority management for quality of service (QoS), especially for real-time IP
- Transaction statistics for bandwidth analysis

1.2 Block Diagram

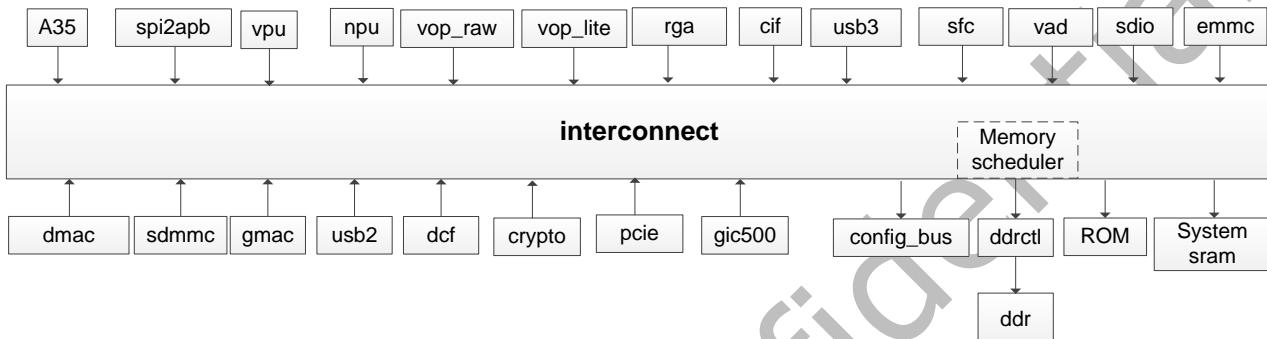


Fig. 1-1 Interconnect diagram

1.3 Function Description

1.3.1 Memory Scheduler

The memory scheduler is an interface to DDR controllers. The scheduler is a special unit case multiplexing several transport interfaces feeding a single memory controller target, with a special arbitration scheme, configurable for optimizing DRAM bandwidth efficiency from the

DDR controller.

The scheduler optimizes DRAM efficiency by choosing the best transaction to execute inside each port input queue: the scheduler can reorder requests across queues, and within a queue as long as they do not have conflicting addresses or ordering constraints. The process for choosing the next transaction to issue to the controller is based on a model of the DRAM behavior (in terms of timings for pages, banks, bank groups, and data bus), a user configuration of arbitration criteria, and a DDR configuration setup representing the current assignment of DRAM banks, bank groups, and page bits to the incoming addresses. It also

makes sure that output queues never overflow.

The scheduler arbitration criteria takes into account the priority characteristics of the transactions, providing a trade-off between QoS requirements of each data flow, and the global DDR efficiency: for example, an urgent read request may take precedence over less urgent write requests, even though there may be a global efficiency loss due to DRAM bus turnaround.

The memory scheduler is connected to third-party DDR controllers, through a simple, ordered interface. A simple DDR controller NIU converts the generic interface output of the scheduler to the appropriate memory controller protocol, and optionally provides access to memory controller registers. It is the responsibility of the controller to fully handle the DRAM state machine.

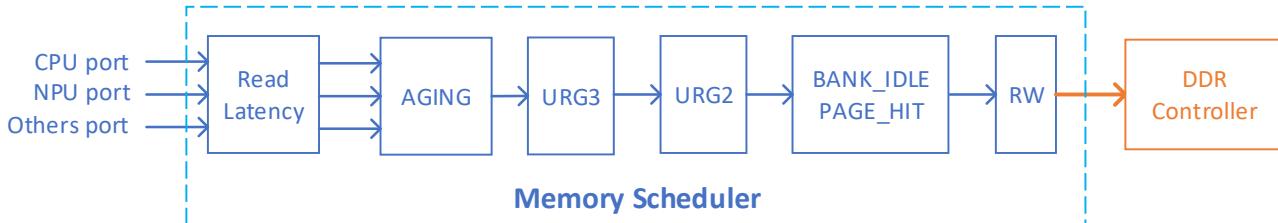


Fig. 1-2 Memory Scheduler diagram

The arbitration criteria of RK1808 is shown above.

Read Latency: The scheduler monitors the latency of read accesses sent to DDR controller, and uses this latency in a feedback loop to control scheduler behavior. When the ceiling is exceeded, the scheduler stalls requests until it obtains the responses of previous reads from the controller. This mechanism thus prevents scheduler-controller misalignment and maintains scheduler-controller system stability, for example, when a DRAM refresh occurs.

AGING: Biases arbitration towards requests that exceed the aging threshold set by register Aging(n) for port (n).

URG3: Biases arbitration in favor of priority 3 requests.

URG2: Biases arbitration in favor of priority 2 requests.

Note: There is a command queue in DDR controller, although the high priority request will be sent to DDR controller first, it does not mean that it can be severed by DDR immediately. The request in DDR controller command queue will be executed in order taking no account of priority.

BANK_IDLE_PAGE_HIT: Biases arbitration in favor of requests whose target bank is idle or whose target row is equal to the active row of the selected bank.

The following method is used to estimate the state of the bank:

Two counters *ActCnt* and *CmdCnt* are associated with each DDR bank, the bank number being obtained from the address using the "B" bits in the selected DDR configuration, set by register 'DeviceConf'.

These counters are decremented at each memory scheduler clock cycle and saturated at 0. A bank is considered idle when both counters are null. The *ActCnt* counter is loaded with *ActToAct* – 1 each time a new page is accessed in that bank.

The *CmdCnt* counter is loaded with:

RdtoMiss + max(*BurstLen*, *N*) – *BurstLen* – 1

each time an *N*-word read is sent, and with

WrtoMiss + max(*BurstLen*, *N*) – 1 each time an *N*-word write is sent.

When the DDR configuration contains bank groups, also known as "G" bits, the following counters are added to each bank group in a device:

CcdLCnt The counter is loaded with the result of *tCCD_L* – 1 each time a new request is granted, and decremented every clock cycle by 1.

RrdCntL The counter is loaded with the result of *tRRD_L* – 1 each time a new page is accessed, and decremented every clock cycle by 1.

RW: Biases arbitration in favor of requests whose RD or WR opcode type matches the type in the last-issued requests.

The criterion is used to estimate the importance of keeping RD and WR transactions together, according to the following method:

Counter *RdWrCnt* and bit *PrvOpc* are used to represent the DDR read-write state. The counter is decremented at each scheduler clock cycle and saturated at 0.

Arbitration is biased in favor of WR when:

PrvOpc = WR or (*PrvOpc* = RD and *RdWrCnt* > *RdToWr*)

The counter is then loaded with:

WrToRr + max(*BurstLen*, *N*) – 1

where *N* is the number of words of the WR request.

Arbitration is biased in favor of RD when:

PrvOpc = RD or (*PrvOpc* = WR and *RdWrCnt* > *WrToRd*)

the counter is then loaded with *WrToRd*+max(*BurstLen*,*N*)-1, where *N* is the number of words of the RD request.

When the DDR configuration contains bank groups, also known as "G" bits, counter

RdWrCntL is added to each device and bank group. The counter is loaded with the result of *RdToWr* + max(*BurstLen*, N) – 1 each time a read request is granted, and with *WrToRdL* + max(*BurstLen*, N) – 1 each time a write is granted. The counter is decremented every clock cycle by 1.

1.3.2 Priority Management

As mentioned above, the memory scheduler will consider the priority of each request to ensure desired quality of service. The priority of each request is configurable through the internal priority register in each master NIU.

The priority registers address of each master NIU and default priority are shown below.

Table 1-1 Internal Priority Setting

| Master NIU | Priority | Register Address |
|-------------|----------|------------------|
| cpu | 2 | 0xfe830008 |
| crypto | 1 | 0xff8d0008 |
| dma | 1 | 0xff8d0088 |
| dcf | 1 | 0xfe820008 |
| gic | 1 | 0xfe840008 |
| npu | 1 | 0xfe850008 |
| vad | 1 | 0xfe810008 |
| emmc | 1 | 0xfe870008 |
| sfc | 1 | 0xfe870088 |
| pcie | 1 | 0xfe880008 |
| gmac | 1 | 0xfe860008 |
| sdio | 1 | 0xfe860088 |
| sdmmc | 1 | 0xfe860108 |
| usb2 | 1 | 0xfe890008 |
| usb3 | 1 | 0xfe890088 |
| isp | 1 | 0xfe8a0008 |
| rga2lite_rd | 1 | 0xfe8a0088 |
| rga2lite_wr | 1 | 0xfe8a0100 |
| cif | 1 | 0xfe8a0188 |
| vop_raw | 3 | 0xfe8b0008 |
| vop_lite | 3 | 0xfe8b0088 |
| vpu | 1 | 0xfe8c0008 |

The priority can be set is 0, 1, 2 or 3. 3 is the highest priority. Set the corresponding register to 0x0 for priority 0, 0x101 for priority 1, 0x202 for priority 2 and 0x303 for priority 3.

Except the internal priority control, *isp*, *cif*, *vop_raw* and *vop_lite* can also change their priority dynamic by their internal register according to buffer status. In order to make this work, the following external enable register must be set to 0x1. In this case, the interconnect will choose the highest priority from the external and internal register. It is better to set the internal priority same as the other master NIUs and external priority higher so that the memory scheduler has more opportunity to optimal DDR efficiency.

Table 1-2 External Enable Register Address

| Master NIU | Register Address |
|------------|------------------|
| isp | 0xfe8a0018 |
| cif | 0xfe8a0118 |
| vop_raw | 0xfe8b0018 |
| vop_lite | 0xfe8b0098 |

1.3.3 Probe

The interconnect provides a service called probe to trace packet and compute traffic statics. There are totally 3 probes to monitor the memory schedule traffic statics and each can be programmed by their register. They are listed below.

Table 1-3 Probe

| Probe Name | Monitor Path | Register Base Address |
|--------------|--|-----------------------|
| cpu_probe | cpu to memory schedule | 0xfe802000 |
| npu_probe | npu to memory schedule | 0xfe802400 |
| others_probe | all the other masters to memory schedule | 0xfe802800 |

Refer to chapter 7.4 for detail register.

1.4 Register Description

1.4.1 Memory Scheduler Registers Summary

Rockchip Confidential

| Name | Offset | Size | Reset Value | Description |
|---------------------------|--------|------|-------------|--|
| <u>MSCH_Id_CoreId</u> | 0x0000 | W | 0x67202902 | Core id |
| <u>MSCH_Id_RevisionId</u> | 0x0004 | W | 0x07711900 | Revision id |
| <u>MSCH_DeviceConf</u> | 0x0008 | W | 0x00000000 | Register DeviceConf stores selectors to the predefined list of DDR configuration to be programmed at initialization phase. The register has two fields, Rank0 and Rank1 |
| <u>MSCH_DeviceSize</u> | 0x000c | W | 0x00000040 | Register DeviceSize is used to set the size of DDR ranks |
| <u>MSCH_DdrTimingA0</u> | 0x0010 | W | 0x20140B13 | Register DdrTimingA(n) of timing register bank n stores timing settings used by memory schedulers to compute bank and page states |
| <u>MSCH_DdrTimingB0</u> | 0x0014 | W | 0x0A030702 | Register DdrTimingB(n) of timing register bank n stores timing settings used by memory schedulers to compute penalties pertaining to bank and page states |
| <u>MSCH_DdrTimingC0</u> | 0x0018 | W | 0x00000002 | Register DdrTimingC(n) of timing register bank n: Stores timing settings used by memory schedulers to compute penalties pertaining to DRAM commands. |
| <u>MSCH_DevToDev0</u> | 0x001c | W | 0x00000221 | Register DevToDev(n) of timing register bank n contains supplementary timing penalties that are incurred when changing data-bus ownership of up to four devices. The penalties are expressed in scheduler clock cycles |
| <u>MSCH_DdrMode</u> | 0x0110 | W | 0x0000000c | Register DdrMode stores the controller behavior description |
| <u>MSCH_Ddr4Timing</u> | 0x0114 | W | 0x00000000 | Long timing for DDR4 Bank Group support |
| <u>MSCH_AgingX0</u> | 0x1000 | W | 0x00000004 | Aging threshold multiplicator |
| <u>MSCH_Aging0</u> | 0x1040 | W | 0x000000FF | AGING slice threshold for port others |
| <u>MSCH_Aging1</u> | 0x1044 | W | 0x000000FF | AGING slice threshold for port npu |
| <u>MSCH_Aging2</u> | 0x1048 | W | 0x000000FF | AGING slice threshold for port cpu |

Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

Base address for memory scheduler is 0xfe800000.

1.4.2 Memory scheduler Detail Register Description

Notes:

In the following tables, tCKD means the clock cycle of DRAM, tCKG means the clock cycle of memory scheduler.

All unit of register related to DDR timing is tCKG.

MSCH_Id_CoreId

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RO | 0x672029 | CoreChecksum Field containing a checksum of the parameters of the IP. |
| 7:0 | RO | 0x02 | CoreTypeId Field identifying the type of IP |

MSCH_Id_RevisionId

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|------------------------|
| 31:0 | RO | 0x07711900 | RevisionId Constant |

MSCH_DeviceConf

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:14 | RO | 0x0 | Reserved |
| 13:8 | RW | 0x00 | Rank1 Rank1 selector of predefined ddrConf configuration The value means the same as Rank0 |
| 7:6 | RO | 0x00 | Reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 5:0 | RW | 0x00 | <p>Rank0 Rank0 selector of predefined ddrConf configuration 6'd0: RRRR RRRR RRRR RRRR RRRB BBCC CCCC CCCC 6'd1: RRRR RRRR RRRR RRRR RRBB BCCC CCCC CCCC 6'd2: RRRR RRRR RRRR RRRR RBBB CCCC CCCC CCCC 6'd3: RRRR RRRR RRRR RRRR BBBC CCCC CCCC CCCC 6'd4: RRRR RRRR RRRR RRRR RRRB BBCC CCCC CCCC CCCC 6'd5: RRRR RRRR RRRR RRRR RRBB CCCC CCCC CCCC 6'd6: RCRR RRRR RRRR RRRR RBBB CCCC CCCC CCCC 6'd7: RRRR RRRR RRRR RRRR RRBB GCCC CCCG CCCC 6'd8: RRRR RRRR RRRR RRRR RBBG CCCC CCCG CCCC 6'd9: RRRR RRRR RRRR RRRR BBGC CCCC CCGC CCCC 6'd10: RRRR RRRR RRRR RRRR RBBC CCCC CCGC CCCC 6'd11: RRRR RRRR RRRR RRRR RRBB CCCC CCCG CCCC 6'd12: RRRR RRRR RRRR RRRR RBBC CCCC CCCC CCCC 6'd13: RRRR RRRR RRRR RRRR RRBB CBCC CCCC CCCC 6'd14: RRRR RRRR RRRR RRRR RRRR BBCC CCCC CCCC 6'd15: RRRR RRRR RRRR RRRR RRRB BCCC CCCC CCCC others: Reserved Here: R: indicates Row bits B: indicates Bank bits C: indicates Column bits and the lowest one or two bits indicates DRAM data width G: indicates Bank Group bits for DDR4</p> |

MSCH_DeviceSize

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:15 | RO | 0x0 | Reserved |
| 14:8 | RW | 0x00 | <p>Rank1 Rank0 size. The granule size is 64MB</p> |
| 7 | RO | 0x0 | Reserved |
| 6:0 | RW | 0x40 | <p>Rank0 Rank0 size. The granule size is 64MB</p> |

MSCH_DdrTimingA0

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:24 | RW | 0x20 | <p>ReadLatency Maximum delay between a read request and the first data response</p> |
| 23:22 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 21:16 | RW | 0x14 | <p>WrToMiss</p> <p>Minimum number of scheduler clock cycles between the last DRAM Write command and a new Read or Write command in another page of the same bank.</p> <p>The field must be set to the following value, rounded to an integer number of scheduler clock cycles:</p> $(WL \times tCkD + tWR + tRP + tRCD) / tCkG$ |
| 15:14 | RO | 0x0 | Reserved |
| 13:8 | RW | 0x0B | <p>RdToMiss</p> <p>Minimum number of scheduler clock cycles between the last DRAM Read command and a new Read or Write command in another page of the same bank.</p> <p>The field must be set to the following value, rounded to an integer number of scheduler clock cycles:</p> $(tRTP + tRP + tRCD - BL \times tCkD / 2) / tCkG$ |
| 7:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x13 | <p>ActToAct</p> <p>Minimum number of scheduler clock cycles between two consecutive DRAM Activate commands on the same bank.</p> <p>The field must be set to the following value, rounded to an integer number of scheduler clock cycles:</p> $tRC / tCkG$ |

MSCH_DdrTimingB0

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:24 | RW | 0x0A | <p>Faw</p> <p>Number of cycles of the FAW period.</p> <p>The field must be set to the following value, rounded to an integer number of scheduler clock cycles:</p> $tFAW / tCkG$ |
| 23:20 | RO | 0x0 | Reserved |
| 19:16 | RW | 0x3 | <p>Rrd</p> <p>Number of cycles between two consecutive Activate commands on different Banks of the same device.</p> <p>The field must be set to the following value, rounded to an integer number of scheduler clock cycles:</p> $tRRD / tCkG$ |
| 15:13 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12:8 | RW | 0x07 | <p>WrToRd Minimum number of scheduler clock cycles between the last DRAM Write command and a Read command. The field must be set to the following value, rounded to an integer of scheduler clock cycles: (WL ×tCkD + tWTR) / tCkG, for DDR2 and DDR3 memories. (WL ×tCkD + tWTR_S) / tCkG, for DDR4 memories.</p> |
| 7:5 | RO | 0x0 | Reserved |
| 4:0 | RW | 0x02 | <p>RdToWr Minimum number of scheduler clock cycles between the last DRAM Read command and a Write command. The field must be set to the following value, rounded to an integer number of scheduler clock cycles: 2 ×tCkD / tCkG, for DDR2 memories. (RL -WL + 2) ×tCkD / tCKG, for DDR3 and DDR4 memories.</p> |

MSCH_DdrTimingC0

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:14 | RO | 0x0 | Reserved |
| 13:8 | RW | 0x00 | <p>WrToMwr Number of scheduler clock cycles between the last write data to the first data of a masked write command on the same bank. This field must be set to 3xBurstPenalty, and must be set to zero for the other DRAM</p> |
| 7:4 | RO | 0x0 | Reserved |
| 3:0 | RW | 0x2 | <p>BurstPenalty DRAM burst duration on the DRAM data bus in scheduler clock cycles. The field must be set to Nd /Ns, where: Nd is the number of DRAM cycles needed to process a DRAM burst of determined size, expressed in bytes. Ns is the minimum number of scheduler cycles to process a DRAM burst of the same size</p> |

MSCH_DevToDev0

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:15 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 14:12 | RW | 0x2 | BusWrToWr number of cycles between the last write data to a device and the first write data of another device. The field must be set according to the third-party DDR controller specification |
| 11 | RO | 0x0 | Reserved |
| 10:8 | RW | 0x2 | BusWrToRd number of cycles between the last write data to a device and the first read data of another device. The field must be set according to the third-party DDR controller specification |
| 7 | RO | 0x0 | Reserved |
| 6:4 | RW | 0x2 | BusRdToWr number of cycles between the last read data of a device and the first write data to another device. The field must be set according to the third-party DDR controller specification |
| 3 | RO | 0x0 | Reserved |
| 2:0 | RW | 0x1 | BusRdToRd number of cycles between the last read data of a device and the first read data of another device. The field must be set according to the third-party DDR controller specification |

MSCH_DdrMode

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | Reserved |
| 23:16 | RO | 0x00 | ForceOrderState ForceOrderState |
| 15:8 | RW | 0x00 | ForceOrder When bit n of register field ForceOrder is set to 1, DRAM commands are executed in the order they arrive at scheduler port n. When field bits are set to 1, and BypassFiltering is also set to 1, command execution order is guaranteed for the corresponding scheduler port |
| 7 | RO | 0x0 | Reserved |
| 6:5 | RW | 0x0 | MwrSize Register MwrSize sets LPDDR4 data width, which is used for masked-write split control. The field must be set to non-zero for LPDDR3 memories. 2'b00: Reserved 2'b01: LPDDR4, 16 bits. 2'b10: LPDDR4, 32 bits. 2'b11: LPDDR4, 32 bits. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 4:3 | RW | 0x1 | BurstSize Register field BurstSize sets the DDR burst size, in bytes, as shown by the following table. 2'b00:16 2'b01:32 2'b10:64 2'b11:128 |
| 2 | RW | 0x1 | FawBank Register field FawBank indicates the number of banks of a given device involved in the FAW period during which four banks can be active. It must be set to 0 for 2-bank memories, and 1 for memories with four banks or more |
| 1 | RW | 0x0 | BypassFiltering When register field BypassFiltering is set to 1, arbiter filters are bypassed and timing register outputs are internally set to an idle value. The field can be useful during DRAM initialization, when training or calibration sequences are performed, and scheduler arbitration is not needed. When the field is set to 0, scheduler arbitration is fully functional, this is the functional usage mode. NOTE: When the field is set to 1, the final arbitration level continues to elect transactions among those presented to the arbiter. Set field ForceOrder to ensure that transactions are executed in order, for instance during DRAM initialization |
| 0 | RW | 0x0 | AutoPrecharge When set to one, pages are automatically closed after each access, when set to zero, pages are left opened until an access in a different page occurs |

MSCH_Ddr4Timing

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:12 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:8 | RW | 0x0 | RrdL Register field RrdL stores the minimum number of scheduler clock cycles on the DRAM command bus, between two Activate commands to different banks of the same bank group in the same device. The filed must be set to tRRD_L/tCkG |
| 7:3 | RW | 0x00 | WrToRdL Register field WrToRdL stores the result of nMin -tData, where nMin is the minimum number of scheduler clock cycles on the DRAM command bus in the interval between a Write command and the next Read command to an already opened page in the same bank group. The filed must be set to (WLxtCkD+tWTR_L)/tCkG |
| 2:0 | RW | 0x0 | CcdL Register field CcdL sets the minimum number of scheduler clock cycles between column access commands to an already opened page in the same bank group. The file must be set to: tCCD_L/tCkG |

MSCH AgingX0

Address: Operational Base + offset (0x1000)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:3 | RO | 0x0 | Reserved |
| 2:0 | RW | 0x4 | AgingX0 Aging threshold multiplicator |

MSCH Aging0

Address: Operational Base + offset (0x1040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0xFF | Aging0 AGING slice threshold for port 0 |

MSCH Aging1

Address: Operational Base + offset (0x1044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0xFF | Aging1 AGING slice threshold for port 1 |

MSCH Aging2

Address: Operational Base + offset (0x1048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0xFF | Aging2 AGING slice threshold for port 2 |

1.4.3 Probe Registers Summary

| Name | Offset | Size | Reset Value | Description |
|-----------------------------|---------------|-------------|--------------------|---|
| <u>PROBE_Id_CoreId</u> | 0x0000 | W | 0x0d867006 | Core id |
| <u>PROBE_Id_RevisionId</u> | 0x0004 | W | 0x0001aa00 | Revision id |
| <u>PROBE_MainCtl</u> | 0x0008 | W | 0x00000000 | Register MainCtl contains probe global control bits |
| <u>PROBE_CfgCtl</u> | 0x000c | W | 0x00000000 | Register CfgCtl contains global enable and active bits. The register, which must be used by software before changing certain packet probe global registers |
| <u>PROBE_StatPeriod</u> | 0x0024 | W | 0x00000000 | Statistics Period |
| <u>PROBE_StatGo</u> | 0x0028 | W | 0x00000000 | Statistics begin control |
| <u>PROBE_Counters_0_Src</u> | 0x0138 | W | 0x00000000 | Register CntSrc indicates the event source used to increment the counter. Unassigned values (non-existing Press level or ExtEvent index, or unimplemented Filter) are equivalent to OFF |
| <u>PROBE_Counters_0_Val</u> | 0x0140 | W | 0x00000000 | Registers Counters_M_Val contain the statistics counter values |
| <u>PROBE_Counters_1_Src</u> | 0x014c | W | 0x00000000 | Register CntSrc indicates the event source used to increment the counter. Unassigned values (non-existing Press level or ExtEvent index, or unimplemented Filter) are equivalent to OFF |
| <u>PROBE_Counters_1_Val</u> | 0x0154 | W | 0x00000000 | Registers Counters_M_Val contain the statistics counter values |
| <u>PROBE_Counters_2_Src</u> | 0x0160 | W | 0x00000000 | Register CntSrc indicates the event source used to increment the counter. Unassigned values (non-existing Press level or ExtEvent index, or unimplemented Filter) are equivalent to OFF |

| Name | Offset | Size | Reset Value | Description |
|----------------------|--------|------|-------------|---|
| PROBE_Counters_2_Val | 0x0168 | W | 0x00000000 | Registers Counters_M_Val contain the statistics counter values |
| PROBE_Counters_3_Src | 0x0174 | W | 0x00000000 | Register CntSrc indicates the event source used to increment the counter. Unassigned values (non-existing Press level or ExtEvent index, or unimplemented Filter) are equivalent to OFF |
| PROBE_Counters_3_Val | 0x017c | W | 0x00000000 | Registers Counters_M_Val contain the statistics counter values |

Notes: **S**ize: **B**- Byte (8 bits) access, **H**W- Half WORD (16 bits) access, **W**-WORD (32 bits) access

1.4.4 Probe Detail Register Description

PROBE_Id_CoreId

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:8 | RO | 0x0d8670 | CoreChecksum Field containing a checksum of the parameters of the IP |
| 7:0 | RO | 0x06 | CoreTypeId Field identifying the type of IP |

PROBE_Id_RevisionId

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|------------------------|
| 31:0 | RO | 0x0001aa00 | RevisionId Constant |

PROBE_MainCtl

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | FiltByteAlwaysChainableEn When set to 0, filters are mapped to all statistic counters when counting bytes or enabled bytes. Therefore, only filter events mapped to even counters can be counted using a pair of chained counters. When set to 1, filters are mapped only to even statistic counters when counting bytes or enabled bytes. Thus, events from any filter can be counted using a pair of chained counters |
| 6 | RO | 0x0 | IntrusiveMode When set to 1, register field IntrusiveMode enables trace operation in Intrusive flow-control mode. When set to 0, the register enables trace operation in Overflow flow-control mode |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 5 | RW | 0x0 | StatCondDump When set, register field StatCondDump enables the dump of a statistics frame to the range of counter values set for registers StatAlarmMin, StatAlarmMax, and AlarmMode. This field also renders register StatAlarmStatus inoperative. When parameter statisticsCounterAlarm is set to False, the StatCondDump register bit is Reserved |
| 4 | RW | 0x0 | AlarmEn When set, register field AlarmEn enables the probe to collect alarm-related information. When the register field bit is null, both TraceAlarm and StatAlarm outputs are driven to 0 |
| 3 | RW | 0x0 | StatEn When set to 1, register field StatEn enables statistics profiling. The probe sends statistics results to the output for signal ObsTx. All statistics counters are cleared when the StatEn bit goes from 0 to 1. When set to 0, counters are disabled |
| 2 | RW | 0x0 | PayloadEn Register field PayloadEn, when set to 1, enables traces to contain headers and payload. When set to 0, only headers are reported |
| 1 | RO | 0x0 | TraceEn Register field TraceEn enables the probe to send filtered packets (Trace) on the ObsTx observation output |
| 0 | RW | 0x0 | ErrEn Register field ErrEn enables the probe to send on the ObsTx output any packet with Error status, independently of filtering mechanisms, thus constituting a simple supplementary global filter |

PROBE CfgCtl

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | Reserved |
| 1 | RO | 0x0 | Active Register field Active is used to inform software that the probe is active. Probe configuration is not allowed during the active state. This bit is raised when bit GlobalEn is set, and is cleared a few cycles after setting GlobalEn to zero (probe is Idle) |
| 0 | RW | 0x0 | GlobalEn Set register field GlobalEn to 1 enable the tracing and statistics collection sub-systems of the packet probe |

PROBE StatPeriod

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | Reserved |
| 4:0 | RW | 0x00 | StatPeriod Register StatPeriod is a 5-bit register that sets a period, within a range of 2 cycles to 2 gigacycles, during which statistics are collected before being dumped automatically. Setting the register implicitly enables automatic mode operation for statistics collection. The period is calculated with the formula: N_Cycle = 2**StatPeriodWhen register StatPeriod is set to its default value 0, automatic dump mode is disabled, and register StatGo is activated for manual mode operation. Note: When parameter statisticsCollection is set to False, StatPeriod is Reserved |

PROBE StatGo

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | StatGo Writing a 1 to the 1-bit pulse register StatGo generates a statistics dump. The register is active when statistics collection operates in manual mode, that is, when register StatPeriod is set to 0. NOTE The written value is not stored in StatGo. A read always returns 0 |

PROBE Counters 0 Src

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:5 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 4:0 | RW | 0x00 | <p>IntEvent Internal packet event</p> <p>5'h00 OFF Counter disabled.</p> <p>5'h01 CYCLE8 Probe clock cycles.</p> <p>5'h02 IDLE Idle cycles during which no packet data is observed.</p> <p>5'h03 XFER Transfer cycles during which packet data is transferred.</p> <p>5'h04 BUSY Busy cycles during which the packet data is made available by the transmitting agent but the receiving agent is not ready to receive it.</p> <p>5'h05 WAIT Wait cycles during a packet in which the transmitting agent suspends the transfer of packet data.</p> <p>5'h06 PKT Packets.</p> <p>5'h08 BYTE Total number of payload bytes.</p> <p>5'h09 PRESS Clock cycles with pressure level > 0.</p> <p>5'h0A PRESS Clock cycles with pressure level > 1.</p> <p>5'h0B PRESS Clock cycles with pressure level > 2.</p> <p>5'h10 CHAIN Carry from counter 2m to counter 2m + 1</p> |

PROBE Counters 0 Val

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RO | 0x0000 | Counters_0_Val Register Val is a read-only register that is always present. The register contains the statistics counter value either pending StatAlarm output, or when statistics collection is suspended subsequent to triggers or signal statSuspend |

PROBE Counters 1 Src

Address: Operational Base + offset (0x014c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:5 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 4:0 | RW | 0x00 | <p>IntEvent Internal packet event Event source type Event description</p> <p>5'h00 OFF Counter disabled. 5'h01 CYCLE8 Probe clock cycles. 5'h02 IDLE Idle cycles during which no packet data is observed. 5'h03 XFER Transfer cycles during which packet data is transferred. 5'h04 BUSY Busy cycles during which the packet data is made available by the transmitting agent but the receiving agent is not ready to receive it. 5'h05 WAIT Wait cycles during a packet in which the transmitting agent suspends the transfer of packet data. 5'h06 PKT Packets. 5'h08 BYTE Total number of payload bytes. 5'h09 PRESS Clock cycles with pressure level > 0. 5'h0A PRESS Clock cycles with pressure level > 1. 5'h0B PRESS Clock cycles with pressure level > 2. 5'h10 CHAIN Carry from counter 2m to counter 2m + 1</p> |

PROBE Counters 1 Val

Address: Operational Base + offset (0x0154)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RO | 0x0000 | Counters_0_Val Register Val is a read-only register that is always present. The register contains the statistics counter value either pending StatAlarm output, or when statistics collection is suspended subsequent to triggers or signal statSuspend |

PROBE Counters 2 Src

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:5 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 4:0 | RW | 0x00 | <p>IntEvent Internal packet event Event source type Event description</p> <p>5'h00 OFF Counter disabled. 5'h01 CYCLE8 Probe clock cycles. 5'h02 IDLE Idle cycles during which no packet data is observed. 5'h03 XFER Transfer cycles during which packet data is transferred. 5'h04 BUSY Busy cycles during which the packet data is made available by the transmitting agent but the receiving agent is not ready to receive it. 5'h05 WAIT Wait cycles during a packet in which the transmitting agent suspends the transfer of packet data. 5'h06 PKT Packets. 5'h08 BYTE Total number of payload bytes. 5'h09 PRESS Clock cycles with pressure level > 0. 5'h0A PRESS Clock cycles with pressure level > 1. 5'h0B PRESS Clock cycles with pressure level > 2. 5'h10 CHAIN Carry from counter 2m to counter 2m + 1</p> |

PROBE Counters 2 Val

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RO | 0x0000 | Counters_0_Val Register Val is a read-only register that is always present. The register contains the statistics counter value either pending StatAlarm output, or when statistics collection is suspended subsequent to triggers or signal statSuspend |

PROBE Counters 3 Src

Address: Operational Base + offset (0x0174)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:5 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 4:0 | RW | 0x00 | <p>IntEvent Internal packet event Event source type Event description</p> <p>5'h00 OFF Counter disabled. 5'h01 CYCLE8 Probe clock cycles. 5'h02 IDLE Idle cycles during which no packet data is observed. 5'h03 XFER Transfer cycles during which packet data is transferred. 5'h04 BUSY Busy cycles during which the packet data is made available by the transmitting agent but the receiving agent is not ready to receive it. 5'h05 WAIT Wait cycles during a packet in which the transmitting agent suspends the transfer of packet data. 5'h06 PKT Packets. 5'h08 BYTE Total number of payload bytes. 5'h09 PRESS Clock cycles with pressure level > 0. 5'h0A PRESS Clock cycles with pressure level > 1. 5'h0B PRESS Clock cycles with pressure level > 2. 5'h10 CHAIN Carry from counter 2m to counter 2m + 1</p> |

PROBE Counters 3 Val

Address: Operational Base + offset (0x017c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RO | 0x0000 | Counters_0_Val Register Val is a read-only register that is always present. The register contains the statistics counter value either pending StatAlarm output, or when statistics collection is suspended subsequent to triggers or signal statSuspend |

1.5 Application Notes

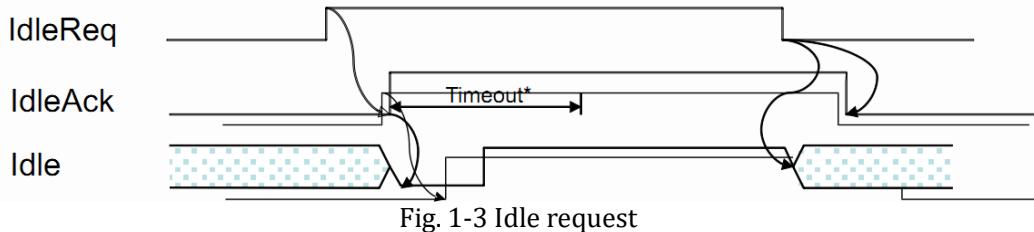
1.5.1 Idle request

The main interconnect supports flushing the ongoing transaction when the software needed to do so.

If the GPU power domain need to disconnect from the main interconnect, Idle request has to be sent to GPU NIU, the NIU will respond a ack, and when it's ready to be disconnect, one Idle signal will be send out. Then, if GPU still have transaction to be sent to the memory scheduler, it will be stalled by the NIU.

If the GPU system power domain is disconnected as the above flow, then CPU want to access to the GPU system, it will response error or held to CPU according to the corresponding grf register setting.

The sequence is like following figure shows:



The idle request is controlled by PMU.

1.5.2 Basic Packet Tracing

To trace packets, the packet probe must be programmed as follows:

- Select the interesting probe listed in section 7.3.3.
- Set field *TraceEn* of register *MainCtl* to 1 to enable forwarding of traced packets to the connected observer. Optionally set field *PayloadEn* of register *MainCtl* to 1 if the packet payload should be included in the trace.
- Set field *GlobalEn* of register *CfgCtl* to 1.

1.5.3 Counting packets over a fixed period

The following programming sequence counts packets at a given probe point using statistic counter 0.

- Select the interesting probe listed in section 7.3.3
- Set field *StatEn* to 1 in register *MainCtl*.
- Set register *Counters_0_Src* to 0x6 (PKT) to count packets.
- Specify the period during which the packets should be counted by setting register *StatPeriod* to: $\log_2(\text{interval expressed in number of probe clock cycles})$
- Set field *GlobalEn* of register *CfgCtl* to 1 to enable packet counting.

Once time $2^{\text{StatPeriod}}$ has elapsed, the number of packets counted is dumped to the observer and can be read from *Counters_0_Val*.

1.5.4 Measuring bandwidth

The following programming sequence example shows how a packet probe can be used to measure bandwidth at a probe point.

Some important points to note about this example are:

- Statistics counters are chained together to support the maximum theoretical bandwidth. Counter 0 is configured to count bytes; counter 1 increments when counter 0 rolls over.
- The counter values are dumped to an observer after time $2^{\text{StatPeriod}}$.

The programming sequence is as follows:

- Select the interesting probe listed in section 7.3.3
- Set register *Counters_0_Src* to 0x8 (BYTES) to count bytes.
- Set register *Counters_1_Src* to 0x10 (CHAIN) to increment when counter 0 wraps.
- Specify the period during which the bytes should be counted by setting register *StatPeriod* to: $\log_2(\text{interval expressed in number of probe clock cycles})$.
- Set field *GlobalEn* of register *CfgCtl* to 1 to enable the counting of bytes.

Once time $2^{\text{StatPeriod}}$ has elapsed, the number of packets counted is dumped to the observer and can be read from *Counters_0_Val* and *Counters_1_Val*.

Chapter 2 Dynamic Memory Interface (DMC)

2.1 Overview

The DMC includes DDR protocol controller (PCTL) and DDR PHY which are a complete memory interface solution for DDR memory subsystems.

The PCTL SoC application bus interface supports AXI interface, with a flexible address mapper logic allow application-specific mapping of row, column, bank, bank group and rank bits to achieve industry leading high-efficiency, low-latency and high-performance from memory interface.

The DDR PHY provides control features to ease the customer implementation of digitally controlled features of the PHY such as initialization, DQS gate training, write leveling training and programmable configuration controls.

The DMC supports the following features:

- Support DDR3/DDR3L/LPDDR2/LPDDR3
- Support up to 2 ranks and up to 2GB capacity
- Support 32-bit, 16-bit DDR data bus width
- Support up to 16-type address mapping
- Support up to 32-bank (including bank group)
- Support DDR burst8 only
- Support different CL/WL latency
- Support DDR3/DDR3L/LPDDR2/LPDDR3 SW frequency change
- Support auto gated clock through DDRC and AXI low power interface
- Support auto put DDR PHY entry or exit self-refresh by DFI lower power interface
- Support auto or SW issue entry or exit clock stop/power-down/self-refresh/deep power-down/max power saving mode
- Support SW or PMU auto let DDR PHY entry or exit retention/self-refresh
- Support open, close, intelligent pre-charge paging policy
- Support advance refresh control
- Support APB interface for PCTL and PHY software-accessible registers
- Support automatic DQS gate training and automatic write leveling training
- Support DDR monitor for debug:
 - AMBA 32-bit APB slave interface
 - Support to monitor DDR read or write address
 - Support to observe whether DDR access address within a specified range
 - Support to do the statistics about DDR read number, write number and active number
 - ◆ Hardware mode
 - ◆ Software mode
 - Support monitor interrupt

2.2 Block Diagram

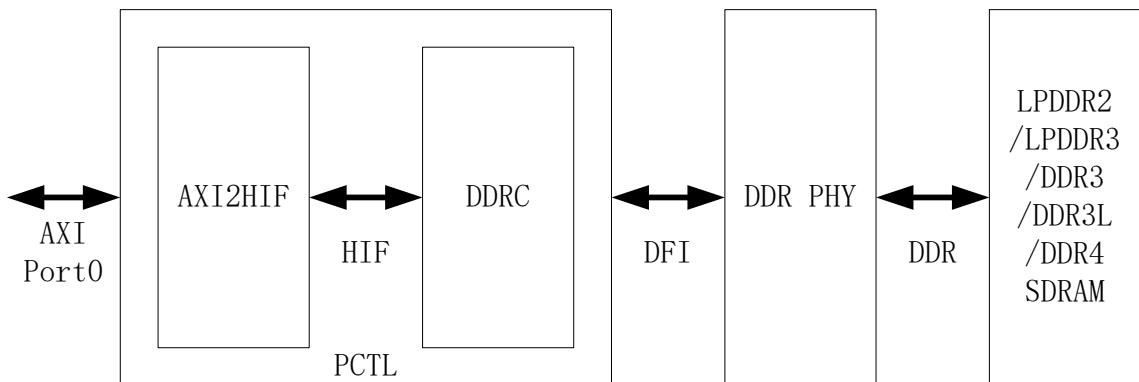


Fig. 2-1 DMC Block Diagram

2.3 Function Description

2.3.1 PCTL

PCTL supports only one AXI Port0, it receives AXI transactions from memory schedule of interconnect. These transactions are queued internally and scheduled for access in order to the SDRAM while satisfying SDRAM protocol timing requirements. It in turn issues commands on the DFI interface to the DDR PHY block which launches and captures data to and from the SDRAM. PCTL contains the following main components:

- AXI2HIF block: This block provides the AXI interface to system level and HIF interface to DDRC block. It provides bus protocol handing, data buffering, data bus size conversion and memory burst alignment. Read is stored in a SRAM, read re-order buffer and return in order to the AXI Port.
- DDRC block: This block issues the read/write commands in order, carries out the DRAM page management, issues DRAM maintenance commands, and implement the DFI interface. Write data is stored in an SRAM until its associated command is issued to the PHY. Read data is handled by the response engine in the DDRC and is returned in order on the HIF.

2.3.2 DDR PHY

DDR PHY supports DDR3/DDR3L/LPDDR2/LPDDR3 SDRAM and provides turnkey physical interface solutions for ICs requiring access to JEDEC compatible SDRAM devices. It is optimized for low power and high speed (up to 1066Mbps for LPDDR2 and up to 1600Mbps for DDR3/DDR3L/LPDDR3) applications with robust timing and small silicon area in 28nm process. It supports all JEDEC DDR3/DDR3L/LPDDR2/LPDDR3 SDRAM components in the market. The PHY components contain DDR specialized functional and utility SSTL and HSUL I/Os up to 2133MHz in 22nm, critical timing synchronization module (TSM) and a low power/jitter DLLs with programmable fine-grain control for any SDRAM interface.

2.3.3 DDR Standby

The function of DDR standby module is to set the standby mode of DDR controller, PHY and memory scheduler. The standby mode is enabled by programming DDR standby control register through APB slave interface. When there are not any bus access to DDR data and register, DDR controller will be idle. Once DDR controller is idle for a period of time, the DDR standby module will generate a low power request to DDR controller and also generate appropriate interface logic to gate the clock of DDR controller, DDR PHY and memory scheduler. It also generates pllpd_dqcmd_if signal to power down the phy pll and clkouten_dqcmd_if signal to disable the output clk of the phy pll.

2.3.4 DDR Monitor

The DDR Monitor Module has three functions, the first function is used when debug, it will

monitor the DDR read or write address. The second function is also used when debug, it will observe whether DDR access address within a specified range. The third function is used to do the statistics about DDR bandwidth and utilization.

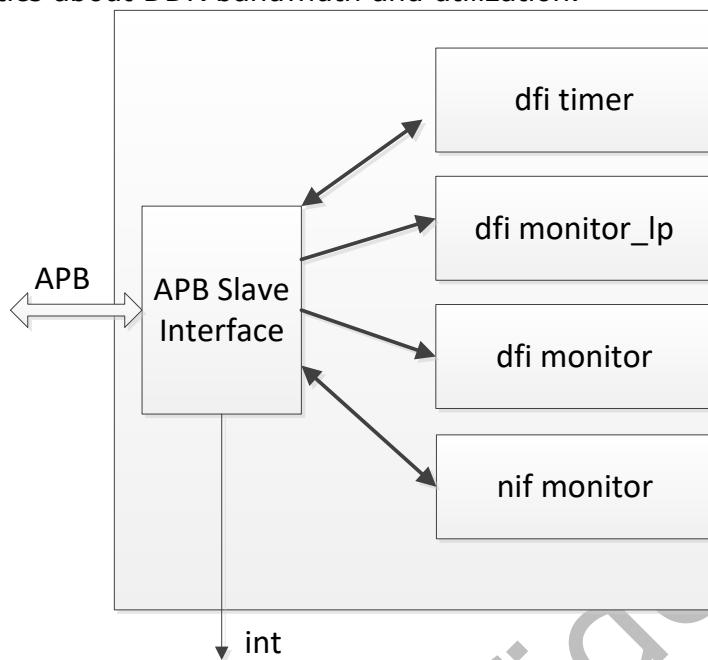


Fig. 2-2 DDR Monitor Block Diagram

The host processor gets access to DDR Monitor Register Block through the APB slave interface with 32-bit bus width, and asserts the active-high level interrupt.

DDR Monitor does the monitor and statistics by **dfi monitor** module, **nif monitor** module and **dfi monitor_lp** module.

2.4 Register Description

Slave address can be divided into different length for different usage, which is shown as follows.

2.4.1 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|-----------------------|--------|------|-------------|---|
| <u>DDRC_MSTR</u> | 0x0000 | W | 0x03040001 | Master Register 0 |
| <u>DDRC_STAT</u> | 0x0004 | W | 0x00000000 | Operating Mode Status Register |
| <u>DDRC_MRCTRL0</u> | 0x0010 | W | 0x00000000 | Mode Register Read/Write Control Register 0 |
| <u>DDRC_MRCTRL1</u> | 0x0014 | W | 0x00000000 | Mode Register Read/Write Control Register 1 |
| <u>DDRC_MRSTAT</u> | 0x0018 | W | 0x00000000 | Mode Register Read/Write Status Register |
| <u>DDRC_MRCTRL2</u> | 0x001c | W | 0x00000000 | Mode Register Read/Write Control Register 2 |
| <u>DDRC_DERATEEN</u> | 0x0020 | W | 0x00000000 | Temperature Derate Enable Register |
| <u>DDRC_DERATEINT</u> | 0x0024 | W | 0x00800000 | Temperature Derate Interval Register |
| <u>DDRC_PWRCTL</u> | 0x0030 | W | 0x00000000 | Lower Power Control Register |

| Name | Offset | Size | Reset Value | Description |
|-----------------|--------|------|-------------|---|
| DDRC_PWRTMG | 0x0034 | W | 0x00402010 | Lower Power Timing Register |
| DDRC_HWLPCCTL | 0x0038 | W | 0x00000003 | Hardware Lower Power Control Register |
| DDRC_HWFFCCTL | 0x003c | W | 0x00000010 | Hardware Fast Frequency Change Control Register |
| DDRC_HWFFCSTAT | 0x0040 | W | 0x00000000 | Hardware Fast Frequency Change Status Register |
| DDRC_RFSHCTL0 | 0x0050 | W | 0x00210000 | Refresh Control Register |
| DDRC_RFSHCTL1 | 0x0054 | W | 0x00000000 | Refresh Control Register 1 |
| DDRC_RFSHCTL3 | 0x0060 | W | 0x00000000 | Refresh Control Register 3 |
| DDRC_RFSHTMG | 0x0064 | W | 0x0062008c | Refresh Timing Register |
| DDRC_CRCPARCTL0 | 0x00c0 | W | 0x00000000 | CRC Parity Control Register 0 |
| DDRC_CRCPARCTL1 | 0x00c4 | W | 0x00001000 | CRC Parity Control Register 1 |
| DDRC_CRCPARSTAT | 0x00cc | W | 0x00000000 | CRC Parity Status Register |
| DDRC_INIT0 | 0x00d0 | W | 0x0002004e | SDRAM Initialization Register 0 |
| DDRC_INIT1 | 0x00d4 | W | 0x00000000 | SDRAM Initialization Register 1 |
| DDRC_INIT2 | 0x00d8 | W | 0x00000d05 | SDRAM Initialization Register 2 |
| DDRC_INIT3 | 0x00dc | W | 0x00000510 | SDRAM Initialization Register 3 |
| DDRC_INIT4 | 0x00e0 | W | 0x00000000 | SDRAM Initialization Register 4 |
| DDRC_INIT5 | 0x00e4 | W | 0x00100004 | SDRAM Initialization Register 5 |
| DDRC_INIT6 | 0x00e8 | W | 0x00000000 | SDRAM Initialization Register 6 |
| DDRC_INIT7 | 0x00ec | W | 0x00000000 | SDRAM Initialization Register 7 |
| DDRC_DIMMCTL | 0x00f0 | W | 0x00000000 | DIMM Control Register |
| DDRC_RANKCTL | 0x00f4 | W | 0x0000066f | Rank Control Register |
| DDRC_DRAMTMG0 | 0x0100 | W | 0x0f101b0f | SDRAM Timing Register 0 |
| DDRC_DRAMTMG1 | 0x0104 | W | 0x00080414 | SDRAM Timing Register 1 |
| DDRC_DRAMTMG2 | 0x0108 | W | 0x0305060d | SDRAM Timing Register 2 |
| DDRC_DRAMTMG3 | 0x010c | W | 0x0050400c | SDRAM Timing Register 3 |
| DDRC_DRAMTMG4 | 0x0110 | W | 0x05040405 | SDRAM Timing Register 4 |
| DDRC_DRAMTMG5 | 0x0114 | W | 0x05050403 | SDRAM Timing Register 5 |
| DDRC_DRAMTMG6 | 0x0118 | W | 0x02020005 | SDRAM Timing Register 6 |
| DDRC_DRAMTMG7 | 0x011c | W | 0x00000202 | SDRAM Timing Register 7 |
| DDRC_DRAMTMG8 | 0x0120 | W | 0x03034405 | SDRAM Timing Register 8 |

| Name | Offset | Size | Reset Value | Description |
|-----------------|--------|------|-------------|--|
| DDRC_DRAMTMG9 | 0x0124 | W | 0x0004040d | SDRAM Timing Register 9 |
| DDRC_DRAMTMG10 | 0x0128 | W | 0x001c180a | SDRAM Timing Register 10 |
| DDRC_DRAMTMG11 | 0x012c | W | 0x440c021c | SDRAM Timing Register 11 |
| DDRC_DRAMTMG12 | 0x0130 | W | 0x00020010 | SDRAM Timing Register 12 |
| DDRC_DRAMTMG13 | 0x0134 | W | 0x1c200004 | SDRAM Timing Register 13 |
| DDRC_DRAMTMG14 | 0x0138 | W | 0x000000a0 | SDRAM Timing Register 14 |
| DDRC_DRAMTMG15 | 0x013c | W | 0x00000000 | SDRAM Timing Register 15 |
| DDRC_DRAMTMG17 | 0x0144 | W | 0x00000000 | SDRAM Timing Register 17 |
| DDRC_ZQCTL0 | 0x0180 | W | 0x02000040 | ZQ Control Register 0 |
| DDRC_ZQCTL1 | 0x0184 | W | 0x02000100 | ZQ Control Register 1 |
| DDRC_ZQCTL2 | 0x0188 | W | 0x00000000 | ZQ Control Register 2 |
| DDRC_ZQSTAT | 0x018c | W | 0x00000000 | ZQ Status Register |
| DDRC_DFITMG0 | 0x0190 | W | 0x07020002 | DFI Timing Register 0 |
| DDRC_DFITMG1 | 0x0194 | W | 0x00000404 | DFI Timing Register 1 |
| DDRC_DFILPCFG0 | 0x0198 | W | 0x07000000 | DFI Lower Power Configuration Register 0 |
| DDRC_DFILPCFG1 | 0x019c | W | 0x00000000 | DFI Lower Power Configuration Register 1 |
| DDRC_DFIUPD0 | 0x01a0 | W | 0x00400003 | DFI Update Register 0 |
| DDRC_DFIUPD1 | 0x01a4 | W | 0x00000001 | DFI Update Register 1 |
| DDRC_DFIUPD2 | 0x01a8 | W | 0x00000001 | DFI Update Register 2 |
| DDRC_DFIMISC | 0x01b0 | W | 0x00000001 | DFI Miscellaneous Control Register |
| DDRC_DFITMG2 | 0x01b4 | W | 0x00000202 | DFI Timing Register 2 |
| DDRC_DFITMG3 | 0x01b8 | W | 0x00000000 | DFI Timing Register 3 |
| DDRC_DFISTAT | 0x01bc | W | 0x00000000 | DFI Status Register |
| DDRC_DBICTL | 0x01c0 | W | 0x00000001 | DM/DBI Control Register |
| DDRC_DFIPHYMSTR | 0x01c4 | W | 0x00000001 | DFI PHY Master |
| DDRC_ADDRMAP0 | 0x0200 | W | 0x00000000 | Address Map Register 0 |
| DDRC_ADDRMAP1 | 0x0204 | W | 0x00000000 | Address Map Register 1 |
| DDRC_ADDRMAP2 | 0x0208 | W | 0x00000000 | Address Map Register 2 |
| DDRC_ADDRMAP3 | 0x020c | W | 0x00000000 | Address Map Register 3 |
| DDRC_ADDRMAP4 | 0x0210 | W | 0x00000000 | Address Map Register 4 |
| DDRC_ADDRMAP5 | 0x0214 | W | 0x00000000 | Address Map Register 5 |
| DDRC_ADDRMAP6 | 0x0218 | W | 0x00000000 | Address Map Register 6 |
| DDRC_ADDRMAP7 | 0x021c | W | 0x00000000 | Address Map Register 7 |
| DDRC_ADDRMAP8 | 0x0220 | W | 0x00000000 | Address Map Register 8 |

| Name | Offset | Size | Reset Value | Description |
|-----------------|--------|------|-------------|--|
| DDRC_ADDRMAP9 | 0x0224 | W | 0x00000000 | Address Map Register 9 |
| DDRC_ADDRMAP10 | 0x0228 | W | 0x00000000 | Address Map Register 10 |
| DDRC_ADDRMAP11 | 0x022c | W | 0x00000000 | Address Map Register 11 |
| DDRC_ODTCFG | 0x0240 | W | 0x04000400 | ODT Configuration Register |
| DDRC_ODTMAP | 0x0244 | W | 0x00002211 | ODT/Rank Map Register |
| DDRC_SCHED | 0x0250 | W | 0x00000804 | Scheduler Control Register |
| DDRC_SCHED1 | 0x0254 | W | 0x00000000 | Scheduler Control Register 1 |
| DDRC_PERFLPR1 | 0x0264 | W | 0x0f00007f | Low Priority Read CAM Register 1 |
| DDRC_PERFWR1 | 0x026c | W | 0x0f00007f | Write CAM Register 1 |
| DDRC_DBG0 | 0x0300 | W | 0x00000000 | Debug Register 0 |
| DDRC_DBG1 | 0x0304 | W | 0x00000000 | Debug Register 1 |
| DDRC_DBGCAM | 0x0308 | W | 0x00000000 | CAM Debug Register |
| DDRC_DBGCMD | 0x030c | W | 0x00000000 | Command Debug Register |
| DDRC_DBGSTAT | 0x0310 | W | 0x00000000 | Status Debug Register |
| DDRC_SWCTL | 0x0320 | W | 0x00000001 | Software Register Programming Control Enable |
| DDRC_SWSTAT | 0x0324 | W | 0x00000001 | Software Register Programming Control Status |
| DDRC_POISONCFG | 0x036c | W | 0x00110011 | AXI Poison Configuration Register |
| DDRC_POISONSTAT | 0x0370 | W | 0x00000000 | AXI Poison Status Register |
| DDRC_PSTAT | 0x03fc | W | 0x00000000 | Port Status Register |
| DDRC_PCCFG | 0x0400 | W | 0x00000000 | Port Common Configuration Register |
| DDRC_PCFG0_0 | 0x0404 | W | 0x00000000 | Port 0 Configuration Read Register |
| DDRC_PCFGW0_0 | 0x0408 | W | 0x00004000 | Port 0 Configuration Write Register |
| DDRC_PCTRL0_0 | 0x0490 | W | 0x00000000 | Port 0 Control Register |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

| Name | Offset | Size | Reset Value | Description |
|-------------|--------|------|-------------|---------------------|
| DDRPHY_REG0 | 0x0000 | W | 0x000000ff | DDR PHY register 00 |
| DDRPHY_REG1 | 0x0004 | W | 0x00000008 | DDR PHY register 01 |
| DDRPHY_REG2 | 0x0008 | W | 0x00000000 | DDR PHY register 02 |
| DDRPHY_REG3 | 0x000c | W | 0x00000022 | DDR PHY register 03 |
| DDRPHY_REG4 | 0x0010 | W | 0x00000022 | DDR PHY register 04 |
| DDRPHY_REG5 | 0x0014 | W | 0x00000000 | DDR PHY register 05 |
| DDRPHY_REG6 | 0x0018 | W | 0x00000002 | DDR PHY register 06 |

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|---------------------|
| DDRPHY_REG9 | 0x0024 | W | 0x00000000 | DDR PHY register 09 |
| DDRPHY_REGA | 0x0028 | W | 0x00000000 | DDR PHY register 0A |
| DDRPHY_REGB | 0x002c | W | 0x00000000 | DDR PHY register 0B |
| DDRPHY_REGC | 0x0030 | W | 0x00000000 | DDR PHY register 0C |
| DDRPHY_REG11 | 0x0044 | W | 0x00000014 | DDR PHY register 11 |
| DDRPHY_REG12 | 0x0048 | W | 0x000000a2 | DDR PHY register 12 |
| DDRPHY_REG13 | 0x004c | W | 0x0000000c | DDR PHY register 13 |
| DDRPHY_REG14 | 0x0050 | W | 0x00000008 | DDR PHY register 14 |
| DDRPHY_REG15 | 0x0054 | W | 0x00000020 | DDR PHY register 15 |
| DDRPHY_REG16 | 0x0058 | W | 0x00000014 | DDR PHY register 16 |
| DDRPHY_REG17 | 0x005c | W | 0x00000044 | DDR PHY register 17 |
| DDRPHY_REG18 | 0x0060 | W | 0x00000014 | DDR PHY register 18 |
| DDRPHY_REG1B | 0x006c | W | 0x00000000 | DDR PHY register 1B |
| DDRPHY_REG1F | 0x007c | W | 0x00000030 | DDR PHY register 1F |
| DDRPHY_REG20 | 0x0080 | W | 0x00000014 | DDR PHY register 20 |
| DDRPHY_REG21 | 0x0084 | W | 0x00000004 | DDR PHY register 21 |
| DDRPHY_REG22 | 0x0088 | W | 0x00000002 | DDR PHY register 22 |
| DDRPHY_REG26 | 0x0098 | W | 0x0000000c | DDR PHY register 26 |
| DDRPHY_REG27 | 0x009c | W | 0x00000000 | DDR PHY register 27 |
| DDRPHY_REG28 | 0x00a0 | W | 0x00000001 | DDR PHY register 28 |
| DDRPHY_REG29 | 0x00a4 | W | 0x00000002 | DDR PHY register 29 |
| DDRPHY_REG2B | 0x00ac | W | 0x00000081 | DDR PHY register 2B |
| DDRPHY_REG2C | 0x00b0 | W | 0x00000000 | DDR PHY register 2C |
| DDRPHY_REG2D | 0x00b4 | W | 0x00000000 | DDR PHY register 2D |
| DDRPHY_REG2E | 0x00b8 | W | 0x00000004 | DDR PHY register 2E |
| DDRPHY_REG2F | 0x00bc | W | 0x00000014 | DDR PHY register 2F |
| DDRPHY_REG30 | 0x00c0 | W | 0x00000014 | DDR PHY register 30 |
| DDRPHY_REG31 | 0x00c4 | W | 0x00000004 | DDR PHY register 31 |
| DDRPHY_REG32 | 0x00c8 | W | 0x00000002 | DDR PHY register 32 |
| DDRPHY_REG36 | 0x00d8 | W | 0x00000004 | DDR PHY register 36 |
| DDRPHY_REG37 | 0x00dc | W | 0x00000000 | DDR PHY register 37 |
| DDRPHY_REG38 | 0x00e0 | W | 0x00000001 | DDR PHY register 38 |
| DDRPHY_REG39 | 0x00e4 | W | 0x00000002 | DDR PHY register 39 |
| DDRPHY_REG3A | 0x00e8 | W | 0x00000001 | DDR PHY register 3A |
| DDRPHY_REG3B | 0x00ec | W | 0x00000081 | DDR PHY register 3B |
| DDRPHY_REG3C | 0x00f0 | W | 0x00000000 | DDR PHY register 3C |
| DDRPHY_REG3D | 0x00f4 | W | 0x00000000 | DDR PHY register 3D |
| DDRPHY_REG3E | 0x00f8 | W | 0x00000004 | DDR PHY register 3E |
| DDRPHY_REG3F | 0x00fc | W | 0x00000014 | DDR PHY register 3F |
| DDRPHY_REG40 | 0x0100 | W | 0x00000014 | DDR PHY register 40 |
| DDRPHY_REG41 | 0x0104 | W | 0x00000004 | DDR PHY register 41 |
| DDRPHY_REG42 | 0x0108 | W | 0x00000002 | DDR PHY register 42 |
| DDRPHY_REG46 | 0x0118 | W | 0x0000000c | DDR PHY register 46 |
| DDRPHY_REG47 | 0x011c | W | 0x00000000 | DDR PHY register 47 |

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|---------------------|
| DDRPHY_REG48 | 0x0120 | W | 0x00000001 | DDR PHY register 48 |
| DDRPHY_REG49 | 0x0124 | W | 0x00000002 | DDR PHY register 49 |
| DDRPHY_REG4A | 0x0128 | W | 0x00000001 | DDR PHY register 4A |
| DDRPHY_REG4B | 0x012c | W | 0x00000091 | DDR PHY register 4B |
| DDRPHY_REG4C | 0x0130 | W | 0x00000000 | DDR PHY register 4C |
| DDRPHY_REG4D | 0x0134 | W | 0x00000000 | DDR PHY register 4D |
| DDRPHY_REG4E | 0x0138 | W | 0x00000004 | DDR PHY register 4E |
| DDRPHY_REG4F | 0x013c | W | 0x00000014 | DDR PHY register 4F |
| DDRPHY_REG50 | 0x0140 | W | 0x00000014 | DDR PHY register 50 |
| DDRPHY_REG51 | 0x0144 | W | 0x00000004 | DDR PHY register 51 |
| DDRPHY_REG52 | 0x0148 | W | 0x00000002 | DDR PHY register 52 |
| DDRPHY_REG56 | 0x0158 | W | 0x0000000c | DDR PHY register 56 |
| DDRPHY_REG57 | 0x015c | W | 0x00000000 | DDR PHY register 57 |
| DDRPHY_REG58 | 0x0160 | W | 0x00000001 | DDR PHY register 58 |
| DDRPHY_REG59 | 0x0164 | W | 0x00000002 | DDR PHY register 59 |
| DDRPHY_REG5A | 0x0168 | W | 0x00000001 | DDR PHY register 5A |
| DDRPHY_REG5B | 0x016c | W | 0x00000091 | DDR PHY register 5B |
| DDRPHY_REG5C | 0x0170 | W | 0x00000000 | DDR PHY register 5C |
| DDRPHY_REG5D | 0x0174 | W | 0x00000000 | DDR PHY register 5D |
| DDRPHY_REG5E | 0x0178 | W | 0x00000004 | DDR PHY register 5E |
| DDRPHY_REG5F | 0x017c | W | 0x00000014 | DDR PHY register 5F |
| DDRPHY_REG70 | 0x01c0 | W | 0x00000077 | DDR PHY register 70 |
| DDRPHY_REG71 | 0x01c4 | W | 0x00000077 | DDR PHY register 71 |
| DDRPHY_REG72 | 0x01c8 | W | 0x00000077 | DDR PHY register 72 |
| DDRPHY_REG73 | 0x01cc | W | 0x00000077 | DDR PHY register 73 |
| DDRPHY_REG74 | 0x01d0 | W | 0x00000077 | DDR PHY register 74 |
| DDRPHY_REG75 | 0x01d4 | W | 0x00000077 | DDR PHY register 75 |
| DDRPHY_REG76 | 0x01d8 | W | 0x00000077 | DDR PHY register 76 |
| DDRPHY_REG77 | 0x01dc | W | 0x00000077 | DDR PHY register 77 |
| DDRPHY_REG78 | 0x01e0 | W | 0x00000077 | DDR PHY register 78 |
| DDRPHY_REG79 | 0x01e4 | W | 0x00000077 | DDR PHY register 79 |
| DDRPHY_REG7A | 0x01e8 | W | 0x00000007 | DDR PHY register 7A |
| DDRPHY_REG7B | 0x01ec | W | 0x00000077 | DDR PHY register 7B |
| DDRPHY_REG7C | 0x01f0 | W | 0x00000077 | DDR PHY register 7C |
| DDRPHY_REG7D | 0x01f4 | W | 0x00000077 | DDR PHY register 7D |
| DDRPHY_REG7E | 0x01f8 | W | 0x00000077 | DDR PHY register 7E |
| DDRPHY_REG7F | 0x01fc | W | 0x00000077 | DDR PHY register 7F |
| DDRPHY_REG80 | 0x0200 | W | 0x00000077 | DDR PHY register 80 |
| DDRPHY_REG81 | 0x0204 | W | 0x00000077 | DDR PHY register 81 |
| DDRPHY_REG82 | 0x0208 | W | 0x00000077 | DDR PHY register 82 |
| DDRPHY_REG83 | 0x020c | W | 0x00000077 | DDR PHY register 83 |
| DDRPHY_REG84 | 0x0210 | W | 0x00000077 | DDR PHY register 84 |
| DDRPHY_REG85 | 0x0214 | W | 0x00000007 | DDR PHY register 85 |
| DDRPHY_REG86 | 0x0218 | W | 0x00000077 | DDR PHY register 86 |

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|---------------------|
| DDRPHY_REG87 | 0x021c | W | 0x00000077 | DDR PHY register 87 |
| DDRPHY_REG88 | 0x0220 | W | 0x00000077 | DDR PHY register 88 |
| DDRPHY_REG89 | 0x0224 | W | 0x00000077 | DDR PHY register 89 |
| DDRPHY_REG8A | 0x0228 | W | 0x00000077 | DDR PHY register 8A |
| DDRPHY_REG8B | 0x022c | W | 0x00000077 | DDR PHY register 8B |
| DDRPHY_REG8C | 0x0230 | W | 0x00000077 | DDR PHY register 8C |
| DDRPHY_REG8D | 0x0234 | W | 0x00000077 | DDR PHY register 8D |
| DDRPHY_REG8E | 0x0238 | W | 0x00000077 | DDR PHY register 8E |
| DDRPHY_REG8F | 0x023c | W | 0x00000077 | DDR PHY register 8F |
| DDRPHY_REG90 | 0x0240 | W | 0x00000007 | DDR PHY register 90 |
| DDRPHY_REG91 | 0x0244 | W | 0x00000077 | DDR PHY register 91 |
| DDRPHY_REG92 | 0x0248 | W | 0x00000077 | DDR PHY register 92 |
| DDRPHY_REG93 | 0x024c | W | 0x00000077 | DDR PHY register 93 |
| DDRPHY_REG94 | 0x0250 | W | 0x00000077 | DDR PHY register 94 |
| DDRPHY_REG95 | 0x0254 | W | 0x00000077 | DDR PHY register 95 |
| DDRPHY_REG96 | 0x0258 | W | 0x00000077 | DDR PHY register 96 |
| DDRPHY_REG97 | 0x025c | W | 0x00000077 | DDR PHY register 97 |
| DDRPHY_REG98 | 0x0260 | W | 0x00000077 | DDR PHY register 98 |
| DDRPHY_REG99 | 0x0264 | W | 0x00000077 | DDR PHY register 99 |
| DDRPHY_REG9A | 0x0268 | W | 0x00000077 | DDR PHY register 9A |
| DDRPHY_REG9B | 0x026c | W | 0x00000007 | DDR PHY register 9B |
| DDRPHY_REGA3 | 0x028c | W | 0x00000030 | DDR PHY register A3 |
| DDRPHY_REGAE | 0x02b8 | W | 0x00000000 | DDR PHY register AE |
| DDRPHY_REGB0 | 0x02c0 | W | 0x00000077 | DDR PHY register B0 |
| DDRPHY_REGB1 | 0x02c4 | W | 0x00000077 | DDR PHY register B1 |
| DDRPHY_REGB2 | 0x02c8 | W | 0x00000077 | DDR PHY register B2 |
| DDRPHY_REGB3 | 0x02cc | W | 0x00000077 | DDR PHY register B3 |
| DDRPHY_REGB4 | 0x02d0 | W | 0x00000077 | DDR PHY register B4 |
| DDRPHY_REGB5 | 0x02d4 | W | 0x00000077 | DDR PHY register B5 |
| DDRPHY_REGB6 | 0x02d8 | W | 0x00000077 | DDR PHY register B6 |
| DDRPHY_REGB7 | 0x02dc | W | 0x00000077 | DDR PHY register B7 |
| DDRPHY_REGB8 | 0x02e0 | W | 0x00000077 | DDR PHY register B8 |
| DDRPHY_REGB9 | 0x02e4 | W | 0x00000077 | DDR PHY register B9 |
| DDRPHY_REGBA | 0x02e8 | W | 0x00000077 | DDR PHY register BA |
| DDRPHY_REGBB | 0x02ec | W | 0x00000077 | DDR PHY register BB |
| DDRPHY_REGBC | 0x02f0 | W | 0x00000077 | DDR PHY register BC |
| DDRPHY_REGBD | 0x02f4 | W | 0x00000077 | DDR PHY register BD |
| DDRPHY_REGBE | 0x02f8 | W | 0x00000077 | DDR PHY register BE |
| DDRPHY_REGC0 | 0x0300 | W | 0x00000077 | DDR PHY register C0 |
| DDRPHY_REGC1 | 0x0304 | W | 0x00000077 | DDR PHY register C1 |
| DDRPHY_REGC2 | 0x0308 | W | 0x00000077 | DDR PHY register C2 |
| DDRPHY_REGC3 | 0x030c | W | 0x00000077 | DDR PHY register C3 |
| DDRPHY_REGC4 | 0x0310 | W | 0x00000077 | DDR PHY register C4 |
| DDRPHY_REGC5 | 0x0314 | W | 0x00000077 | DDR PHY register C5 |

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|---------------------|
| DDRPHY_REGC6 | 0x0318 | W | 0x00000077 | DDR PHY register C6 |
| DDRPHY_REGC7 | 0x031c | W | 0x00000077 | DDR PHY register C7 |
| DDRPHY_REGC8 | 0x0320 | W | 0x00000077 | DDR PHY register C8 |
| DDRPHY_REGC9 | 0x0324 | W | 0x00000077 | DDR PHY register C9 |
| DDRPHY_REGCA | 0x0328 | W | 0x00000007 | DDR PHY register CA |
| DDRPHY_REGCB | 0x032c | W | 0x00000077 | DDR PHY register CB |
| DDRPHY_REGCC | 0x0330 | W | 0x00000077 | DDR PHY register CC |
| DDRPHY_REGCD | 0x0334 | W | 0x00000077 | DDR PHY register CD |
| DDRPHY_REGCE | 0x0338 | W | 0x00000077 | DDR PHY register CE |
| DDRPHY_REGCF | 0x033c | W | 0x00000077 | DDR PHY register CF |
| DDRPHY_REGD0 | 0x0340 | W | 0x00000077 | DDR PHY register D0 |
| DDRPHY_REGD1 | 0x0344 | W | 0x00000077 | DDR PHY register D1 |
| DDRPHY_REGD2 | 0x0348 | W | 0x00000077 | DDR PHY register D2 |
| DDRPHY_REGD3 | 0x034c | W | 0x00000077 | DDR PHY register D3 |
| DDRPHY_REGD4 | 0x0350 | W | 0x00000077 | DDR PHY register D4 |
| DDRPHY_REGD5 | 0x0354 | W | 0x00000007 | DDR PHY register D5 |
| DDRPHY_REGD6 | 0x0358 | W | 0x00000077 | DDR PHY register D6 |
| DDRPHY_REGD7 | 0x035c | W | 0x00000077 | DDR PHY register D7 |
| DDRPHY_REGD8 | 0x0360 | W | 0x00000077 | DDR PHY register D8 |
| DDRPHY_REGD9 | 0x0364 | W | 0x00000077 | DDR PHY register D9 |
| DDRPHY_REGDA | 0x0368 | W | 0x00000077 | DDR PHY register DA |
| DDRPHY_REGDB | 0x036c | W | 0x00000077 | DDR PHY register DB |
| DDRPHY_REGDC | 0x0370 | W | 0x00000077 | DDR PHY register DC |
| DDRPHY_REGDD | 0x0374 | W | 0x00000077 | DDR PHY register DD |
| DDRPHY_REGDE | 0x0378 | W | 0x00000070 | DDR PHY register DE |
| DDRPHY_REGDF | 0x037c | W | 0x00000077 | DDR PHY register DF |
| DDRPHY_REGE0 | 0x0380 | W | 0x00000007 | DDR PHY register E0 |
| DDRPHY_REGE1 | 0x0384 | W | 0x00000077 | DDR PHY register E1 |
| DDRPHY_REGE2 | 0x0388 | W | 0x00000077 | DDR PHY register E2 |
| DDRPHY_REGE3 | 0x038c | W | 0x00000077 | DDR PHY register E3 |
| DDRPHY_REGE4 | 0x0390 | W | 0x00000077 | DDR PHY register E4 |
| DDRPHY_REGE5 | 0x0394 | W | 0x00000077 | DDR PHY register E5 |
| DDRPHY_REGE6 | 0x0398 | W | 0x00000077 | DDR PHY register E6 |
| DDRPHY_REGE7 | 0x039c | W | 0x00000077 | DDR PHY register E7 |
| DDRPHY_REGE8 | 0x03a0 | W | 0x00000077 | DDR PHY register E8 |
| DDRPHY_REGE9 | 0x03a4 | W | 0x00000077 | DDR PHY register E9 |
| DDRPHY_REGEA | 0x03a8 | W | 0x00000077 | DDR PHY register EA |
| DDRPHY_REGEB | 0x03ac | W | 0x00000007 | DDR PHY register EB |
| DDRPHY_REGEC | 0x03b0 | W | 0x00000000 | DDR PHY register EC |
| DDRPHY_REGF0 | 0x03c0 | W | 0x00000000 | DDR PHY register F0 |
| DDRPHY_REGF1 | 0x03c4 | W | 0x00000000 | DDR PHY register F1 |
| DDRPHY_REGF2 | 0x03c8 | W | 0x00000000 | DDR PHY register F2 |
| DDRPHY_REGF3 | 0x03cc | W | 0x00000000 | DDR PHY register F3 |
| DDRPHY_REGF4 | 0x03d0 | W | 0x00000000 | DDR PHY register F4 |

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|---------------------|
| DDRPHY_REGFA | 0x03e8 | W | 0x00000000 | DDR PHY register FA |
| DDRPHY_REGFB | 0x03ec | W | 0x00000000 | DDR PHY register FB |
| DDRPHY_REGFC | 0x03f0 | W | 0x00000000 | DDR PHY register FC |
| DDRPHY_REGFD | 0x03f4 | W | 0x00000000 | DDR PHY register FD |
| DDRPHY_REGFE | 0x03f8 | W | 0x00000000 | DDR PHY register FE |
| DDRPHY_REGFF | 0x03fc | W | 0x00000000 | DDR PHY register FF |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

2.4.2 Detail Register Description

DDRC_MSTR

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:30 | RW | 0x0 | device_config Configuration of device used 00: x4 device 01: x8 device 10: x16 device 11: x32 device Programming Mode: Static |
| 29 | RW | 0x0 | frequency_mode Choose which register are used 0: Original registers 1: FREQ1 registers Programming Mode: Quasi-dynamic Group 2 |
| 28:26 | RO | 0x0 | reserved |
| 25:24 | RW | 0x3 | active_ranks 01: One Rank 11: Two Ranks Programming Mode: Static |
| 23 | RO | 0x0 | reserved |
| 22 | RW | 0x0 | frequency_ratio selects the frequency ratio 0: 1:2 mode 1: 1:1 mode Programming Mode: Static |
| 21:20 | RW | 0x0 | reserved |
| 19:16 | RW | 0x4 | burst_rdwr SDRAM burst length used: 0001: burst length of 2 0010: burst length of 4 0100: burst length of 8 1000: burst length of 16 Programming Mode: Static |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 15 | RW | 0x0 | <p>dll_off_mode 1: dll-off mode for lower frequency operation 0: dll-on mode for normal frequency operation Programming Mode: Quasi-dynamic Group 2</p> |
| 14 | RO | 0x0 | reserved |
| 13:12 | RW | 0x0 | <p>data_bus_width 00: full DQ bus width to SDRAM 01: half DQ bus width to SDRAM 1x: reserved Programming Mode: Static</p> |
| 11 | RW | 0x0 | <p>geardown_mode 1: enable geardown mode 0: normal mode Programming Mode: Quasi-dynamic Group 2</p> |
| 10 | RW | 0x0 | <p>en_2t_timing_mode 1: use 2T timing 0: use 1T timing Programming Mode: Static</p> |
| 9 | RW | 0x0 | <p>burstchop 1: enable burst-chop Programming Mode: Static</p> |
| 8 | RW | 0x0 | <p>burst_mode 0: sequential burst mode 1: interleaved burst mode Programming Mode: Static</p> |
| 7:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | reserved |
| 4 | RW | 0x0 | reserved |
| 3 | RW | 0x0 | <p>lpddr3 1: LPDDR3 0: Non-LPDDR3 Programming Mode: Static</p> |
| 2 | RW | 0x0 | <p>lpddr2 1: LPDDR2 0: Non-LPDDR2 Programming Mode: Static</p> |
| 1 | RW | 0x0 | <p>mobile 1: Mobile/LPDDR 0: Non-Mobile Programming Mode: Static</p> |
| 0 | RW | 0x1 | <p>ddr3 1: DDR3 0: Non-DDR3 Programming Mode: Static</p> |

DDRC_STAT

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | reserved |
| 12 | RO | 0x0 | selfref_cam_not_empty Self refresh with CAMs not empty. Set to 1 when Self Refresh is entered but CAMs are not drained. Cleared after exiting Self Refresh Programming Mode: Dynamic |
| 11:10 | RO | 0x0 | reserved |
| 9:8 | RO | 0x0 | reserved |
| 7:6 | RO | 0x0 | reserved |
| 5:4 | RO | 0x0 | selfref_type Flags if Self Refresh or SR-Powerdown is entered and if it was under Automatic Self Refresh control only or not. 00:SDRAM is not in Self Refresh or SR-Powerdown. If retry is enabled by CRCPARCTRL1.crc_parity_retry_enable, this also indicates SRE command is still in parity error window or retry is in-progress. 11:SDRAM is in Self Refresh or SRPowerdown (LPDDR), which was caused by Automatic Self Refresh only. If retry is enabled, this guarantees SRE command is executed correctly without parity error. 10:SDRAM is in Self Refresh or SRPowerdown (LPDDR), which was not caused solely under Automatic Self Refresh control. It could have been caused by Hardware Low Power Interface and/or Software (PWRCTL.selfref_sw). If retry is enabled, this guarantees SRE command is executed correctly without parity error. 01: SDRAM is in Self Refresh, which was caused by PHY Master Request. Programming Mode: Dynamic |
| 3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | operating_mode Operating mode. This is 3-bits wide in configurations with mDDR/LPDDR2/LPDDR3 support and 2-bits in all other configurations. non-mDDR/LPDDR2/LPDDR3 and non-DDR designs: 00: Init; 01:Normal; 10:Power-down; 11:Self refresh mDDR/LPDDR2/LPDDR3 designs: 000:Init; 001:Normal; 010:Power-down; 011:Self refresh; 1XX:Deep power-down / Maximum Power Saving Mode Programming Mode: Dynamic |

DDRC_MRCTRL0

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31 | RW | 0x0 | <p>mr_wr</p> <p>Setting this register bit to 1 triggers a mode register read or write operation. When the MR operation is complete, the uMCTL2 automatically clears this bit. The other register fields of this register must be written in a separate APB transaction, before setting this mr_wr bit. It is recommended NOT to set this signal if in Init, Deep power-down or MPSM operating modes.</p> <p>Programming Mode: Dynamic</p> |
| 30 | RW | 0x0 | <p>pba_mode</p> <p>Indicates whether PBA access is executed. When setting this bit to 1 along with setting pda_en to 1, uMCTL2 initiates PBA access instead of PDA access.</p> <p>0: Per DRAM Addressability mode; 1: Per Buffer Addressability mode</p> <p>The completion of PBA access is confirmed by MRSTAT.pda_done in the same way as PDA.</p> <p>Programming Mode: Dynamic</p> |
| 29:16 | RO | 0x0 | reserved |
| 15:12 | RW | 0x0 | <p>mr_addr</p> <p>Address of the mode register that is to be written to.</p> <p>0000: MR0; 0001: MR1; 0010: MR2; 0011: MR3; 0100: MR4; 0101: MR5; 0110: MR6; 0111: MR7.</p> <p>Don't Care for LPDDR2/LPDDR3 (see MRCTRL1.mr_data for mode register addressing in LPDDR2/LPDDR3)</p> <p>Programming Mode: Dynamic</p> |
| 11:6 | RO | 0x0 | reserved |
| 5:4 | RW | 0x0 | <p>mr_rank</p> <p>Controls which rank is accessed by MRCTRL0.mr_wr. Normally, it is desired to access all ranks, so all bits should be set to 1.</p> <p>0x1: select rank 0 only</p> <p>0x2: select rank 1 only</p> <p>0x3: select rank 0 and 1</p> <p>Programming Mode: Dynamic</p> |
| 3 | RW | 0x0 | <p>sw_init_int</p> <p>Indicates whether Software intervention is allowed via MRCTRL0/MRCTRL1 before automatic SDRAM initialization routine or not.</p> <p>Note that this must be cleared to 0 after completing Software operation. Otherwise, SDRAM initialization routine will not restart.</p> <p>0: Software intervention is not allowed; 1: Software intervention is allowed</p> <p>Programming Mode: Dynamic</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 2 | RW | 0x0 | pda_en Indicates whether the mode register operation is MRS in PDA mode or not 0: MRS; 1: MRS in Per DRAM Addressability mode Note that when pba_mode=1, PBA access is initiated instead of PDA access. Programming Mode: Dynamic |
| 1 | RW | 0x0 | reserved |
| 0 | RW | 0x0 | mr_type Indicates whether the mode register operation is read or write. Only used for LPDDR2/LPDDR3. 0: Write; 1: Read Programming Mode: Dynamic |

DDRC_MRCTRL1

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:18 | RO | 0x0 | reserved |
| 17:0 | RW | 0x00000 | mr_data Mode register write data for all non-LPDDR2/non-LPDDR3 modes. For LPDDR2/LPDDR3, MRCTRL1[15:0] are interpreted as [15:8] MR Address [7:0] MR data for writes, don't care for reads. Programming Mode: Dynamic |

DDRC_MRSTAT

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:9 | RO | 0x0 | reserved |
| 8 | RO | 0x0 | <p>pda_done</p> <p>The SoC core may initiate a MR write operation in PDA/PBA mode only if this signal is low. This signal goes high when three consecutive MRS commands related to the PDA/PBA mode are issued to the SDRAM. This signal goes low when MRCTRL0.pda_en becomes 0. Therefore, it is recommended to write MRCTRL0.pda_en to 0 after this signal goes high in order to prepare to perform PDA operation next time.</p> <p>0: Indicates that mode register write operation related to PDA/PBA is in progress or has not started yet.</p> <p>1: Indicates that mode register write operation related to PDA/PBA has completed.</p> <p>Programming Mode: Dynamic</p> |
| 7:1 | RO | 0x0 | reserved |
| 0 | RO | 0x0 | <p>mr_wr_busy</p> <p>The SoC core may initiate a MR write operation only if this signal is low. This signal goes high in the clock after the uMCTL2 accepts the MRW/MRR request. It goes low when the MRW/MRR command is issued to the SDRAM. It is recommended not to perform MRW/MRR commands when MRSTAT.mr_wr_busy is high.</p> <p>0: Indicates that the SoC core can initiate a mode register write operation</p> <p>1: Indicates that mode register write operation is in progress</p> <p>Programming Mode: Dynamic</p> |

DDRC MRCTRL2

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>mr_device_sel</p> <p>Indicates the device(s) to be selected during the MRS that happens in PDA mode. Each bit is associated with one device. For example, bit[0] corresponds to Device 0, bit[1] to Device 1 etc.</p> <p>A '1' should be programmed to indicate that the MRS command should be applied to that device.</p> <p>A '0' indicates that the MRS commands should be skipped for that device.</p> <p>Programming Mode: Dynamic</p> |

DDRC DERATEEN

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | reserved |
| 10:8 | RW | 0x0 | reserved |
| 7:4 | RW | 0x0 | derate_byte Derate byte Present only in designs configured to support LPDDR2/LPDDR3 Indicates which byte of the MRR data is used for derating. Programming Mode: Static |
| 3:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | derate_value Derate value 0: Derating uses +1. 1: Derating uses +2. Present only in designs configured to support LPDDR2/LPDDR3 Set to 0 for all LPDDR2 speed grades as derating value of +1.875 ns is less than a core_ddrc_core_clk period. For LPDDR3/4, if the period of core_ddrc_core_clk is less than 1.875ns, this register field should be set to 1; otherwise it should be set to 0. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 0 | RW | 0x0 | derate_enable Enables derating 0: Timing parameter derating is disabled 1: Timing parameter derating is enabled using MR4 read value. Present only in designs configured to support LPDDR2/LPDDR3 This field must be set to 0 for non-LPDDR2/LPDDR3 mode. Programming Mode: Dynamic |

DDRC DERATEINT

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00800000 | mr4_read_interval Interval between two MR4 reads, used to derate the timing parameters. Present only in designs configured to support LPDDR2/LPDDR3. This register must not be set to zero. Unit: DFI clock cycle. Programming Mode: Static |

DDRC PWRCTL

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 7 | RW | 0x0 | <p>dis_cam_drain_selfref Indicates whether skipping CAM draining is allowed when entering Self-Refresh.</p> <p>This register field cannot be modified while PWRCTL.selfref_sw==1.</p> <p>0: CAMs must be empty before entering SR 1: CAMs are not emptied before entering SR</p> <p>Programming Mode: Dynamic</p> |
| 6 | RW | 0x0 | <p>stay_in_selfref Self refresh state is an intermediate state to enter to Selfrefresh power down state or exit Self refresh power down state for LPDDR.</p> <p>This register controls transition from the Self refresh state.</p> <p>1: Prohibit transition from Self refresh state 0: Allow transition from Self refresh state</p> <p>Programming Mode: Dynamic</p> |
| 5 | RW | 0x0 | <p>selfref_sw A value of 1 to this register causes system to move to Self Refresh state immediately, as long as it is not in INIT or DPD/MPSM operating_mode. This is referred to as Software Entry/Exit to Self Refresh.</p> <p>1: Software Entry to Self Refresh 0: Software Exit from Self Refresh</p> <p>Programming Mode: Dynamic</p> |
| 4 | RW | 0x0 | reserved |
| 3 | RW | 0x0 | <p>en_dfi_dram_clk_disable Enable the assertion of dfi_dram_clk_disable whenever a clock is not required by the SDRAM. If set to 0, dfi_dram_clk_disable is never asserted.</p> <p>Assertion of dfi_dram_clk_disable is as follows: In DDR2/DDR3, can only be asserted in Self Refresh. In mDDR/LPDDR2/LPDDR3, can be asserted in the Self Refresh, Power Down, Deep Power Down and Normal operation (Clock Stop)</p> <p>Programming Mode: Dynamic</p> |
| 2 | RW | 0x0 | <p>deeppowerdown_en When this is 1, uMCTL2 puts the SDRAM into deep powerdown mode when the transaction store is empty.</p> <p>This register must be reset to '0' to bring uMCTL2 out of deep power-down mode. Controller performs automatic SDRAM initialization on deep power-down exit.</p> <p>Present only in designs configured to support mDDR or LPDDR2 or LPDDR3. For non-mDDR/non-LPDDR2/nonLPDDR3, this register should not be set to 1.</p> <p>Programming Mode: Dynamic</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 1 | RW | 0x0 | <p>powerdown_en</p> <p>If true then the uMCTL2 goes into power-down after a programmable number of cycles "maximum idle clocks before power down" PWRTMG.powerdown_to_x32).</p> <p>This register bit may be re-programmed during the course of normal operation.</p> <p>Programming Mode: Dynamic</p> |
| 0 | RW | 0x0 | <p>selfref_en</p> <p>If true then the uMCTL2 puts the SDRAM into Self Refresh after a programmable number of cycles "maximum idle clocks before Self Refresh (PWRTMG.selfref_to_x32)". This register bit may be re-programmed during the course of normal operation.</p> <p>Programming Mode: Dynamic</p> |

DDRC PWRTMG

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x40 | <p>selfref_to_x32</p> <p>After this many clocks of the DDRC command channel being idle the uMCTL2 automatically puts the SDRAM into Self Refresh. The DDRC command channel is considered idle when there are no HIF commands outstanding. This must be enabled in the PWRCTL.selfref_en.</p> <p>Unit: Multiples of 32 DFI clocks.</p> <p>Programming Mode: Static</p> |
| 15:8 | RW | 0x20 | <p>t_dpd_x4096</p> <p>Minimum deep power-down time.</p> <p>For mDDR, value from the JEDEC specification is 0 as mDDR exits from deep power-down mode immediately after PWRCTL.deeppowerdown_en is de-asserted.</p> <p>For LPDDR2/LPDDR3, value from the JEDEC specification is 500us.</p> <p>Unit: Multiples of 4096 DFI clocks.</p> <p>Programming Mode: Static</p> |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x10 | <p>powerdown_to_x32</p> <p>After this many clocks of the DDRC command channel being idle the uMCTL2 automatically puts the SDRAM into powerdown. The DDRC command channel is considered idle when there are no HIF commands outstanding. This must be enabled in the PWRCTL.powerdown_en.</p> <p>Unit: Multiples of 32 DFI clocks</p> <p>Programming Mode: Static</p> |

DDRC_HWLPCtl

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x000 | <p>hw_lp_idle_x32 Hardware idle period. The cactive_ddrc output is driven low if the DDRC command channel is idle for hw_lp_idle * 32 cycles if not in INIT or DPD/MPSM operating_mode. The DDRC command channel is considered idle when there are no HIF commands outstanding. The hardware idle function is disabled when hw_lp_idle_x32=0. Unit: Multiples of 32 DFI clocks.</p> <p>Programming Mode: Static</p> |
| 15:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | <p>hw_lp_exit_idle_en When this bit is programmed to 1 the cactive_in_ddrc pin of the DDRC can be used to exit from the automatic clock stop, automatic power down or automatic self-refresh modes. Note, it will not cause exit of Self-Refresh that was caused by Hardware Low Power Interface and/or Software (PWRCTL.selfref_sw).</p> <p>Programming Mode: Static</p> |
| 0 | RW | 0x1 | <p>hw_lp_en Enable for Hardware Low Power Interface.</p> <p>Programming Mode: Quasi-dynamic Group 3</p> |

DDRC_HWFFCCTL

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | <p>target_vrcg Set target value of VRCG (MR13 OP[3]). This field value is used when HWFFC request has been issued.</p> <p>Programming Mode: Static</p> |
| 5 | RW | 0x0 | <p>init_vrcg Set initial value of VRCG (MR13 OP[3]). This field value is used when HWFFCCTL.hwffc_en has been changed to 11.</p> <p>Programming Mode: Static</p> |
| 4 | RW | 0x1 | <p>init_fsp Set initial value of FSP-OP (MR13 OP[7]). This field value is used when HWFFCCTL.hwffc_en has been changed to 11.</p> <p>Programming Mode: Static</p> |
| 3:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 0 | RW | 0x0 | <p>hwffc_en Enable HWFFC through Hardware Low Power Interface. The other fields of this register is used only when changing this field to 11.</p> <p>00: Disable HWFFC 10: Intermediate, set only when disabling HWFFC 11: Enable HWFFC 01: Not allowed Programming Mode: Dynamic</p> |

DDRC HWFFCSTAT

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | reserved |
| 9 | RO | 0x0 | <p>current_vrcg Indicates current value of VRCG (MR13 OP[3]). Programming Mode: Dynamic</p> |
| 8 | RO | 0x0 | <p>current_fsp Indicates current value of FSP-OP (MR13 OP[7]). Programming Mode: Dynamic</p> |
| 7:5 | RO | 0x0 | reserved |
| 4 | RO | 0x0 | <p>current_frequency Indicates the current frequency. 0: Frequency 0/Normal 1: Frequency 1/FREQ1 Programming Mode: Dynamic</p> |
| 3:2 | RO | 0x0 | reserved |
| 1 | RO | 0x0 | <p>hwffc_operating_mode Operating mode of HWFFC. 0: Normal 1: Self Refresh or SR-Powerdown Programming Mode: Dynamic</p> |
| 0 | RO | 0x0 | <p>hwffc_in_progress Indicates HWFFC is in progress. Programming Mode: Dynamic</p> |

DDRC RFSHCTL0

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:24 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23:20 | RW | 0x2 | <p>refresh_margin</p> <p>Threshold value in number of DFI clock cycles before the critical refresh or page timer expires. A critical refresh is to be issued before this threshold is reached. It is recommended that this not be changed from the default value, currently shown as 0x2. It must always be less than internally used t_rfc_nom_x32. Note that, in LPDDR2/LPDDR3, internally used t_rfc_nom_x32 may be equal to RFSHTMG.t_rfc_nom_x32>>2 if derating is enabled (DERATEEN.derate_enable=1). Otherwise, internally used t_rfc_nom_x32 will be equal to RFSHTMG.t_rfc_nom_x32.</p> <p>Unit: Multiples of 32 DFI clocks.</p> <p>Programming Mode: Dynamic - Refresh Related</p> |
| 19:17 | RO | 0x0 | reserved |
| 16:12 | RW | 0x10 | <p>refresh_to_x32</p> <p>If the refresh timer (tRFCnom, also known as tREFI) has expired at least once, but it has not expired (RFSHCTL0.refresh_burst+1) times yet, then a speculative refresh may be performed. A speculative refresh is a refresh performed at a time when refresh would be useful, but before it is absolutely required. When the SDRAM bus is idle for a period of time determined by this RFSHCTL0.refresh_to_x32 and the refresh timer has expired at least once since the last refresh, then a speculative refresh is performed. Speculative refreshes continues successively until there are no refreshes pending or until new reads or writes are issued to the uMCTL2.</p> <p>Unit: Multiples of 32 DFI clocks.</p> <p>Programming Mode: Dynamic - Refresh Related</p> |
| 11:9 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 8:4 | RW | 0x00 | <p>refresh_burst The programmed value + 1 is the number of refresh timeouts that is allowed to accumulate before traffic is blocked and the refreshes are forced to execute. Closing pages to perform a refresh is a one-time penalty that must be paid for each group of refreshes. Therefore, performing refreshes in a burst reduces the per-refresh penalty of these page closings. Higher numbers for RFSHCTL.refresh_burst slightly increases utilization; lower numbers decreases the worstcase latency associated with refreshes.</p> <p>0: single refresh 1: burst-of-2 refresh 7: burst-of-8 refresh</p> <p>For DDR2/3, the refresh is always per-rank and not perbank. The rank refresh can be accumulated over $8*tREFI$ cycles using the burst refresh feature. If using PHY-initiated updates, care must be taken in the setting of RFSHCTL0.refresh_burst, to ensure that tRFCmax is not violated due to a PHY-initiated update occurring shortly before a refresh burst was due. In this situation, the refresh burst will be delayed until the PHYinitiated update is complete.</p> <p>Programming Mode: Dynamic - Refresh Related</p> |
| 3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | <p>per_bank_refresh 1: Per bank refresh; 0: All bank refresh. Per bank refresh allows traffic to flow to other banks. Per bank refresh is not supported by all LPDDR2 devices but should be supported by all LPDDR3 devices.</p> <p>Programming Mode: Static</p> |
| 1:0 | RO | 0x0 | reserved |

DDRC_RFSHCTL1

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x000 | <p>refresh_timer1_start_value_x32 Refresh timer start for rank 1 (only present in multi-rank configurations). This is useful in staggering the refreshes to multiple ranks to help traffic to proceed. This is explained in Refresh Controls section of architecture chapter. Unit: Multiples of 32 DFI clock cycles.</p> <p>Programming Mode: Dynamic - Refresh Related</p> |
| 15:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 11:0 | RW | 0x000 | <p>refresh_timer0_start_value_x32 Refresh timer start for rank 0 (only present in multi-rank configurations). This is useful in staggering the refreshes to multiple ranks to help traffic to proceed. This is explained in Refresh Controls section of architecture chapter. Unit: Multiples of 32 DFI clock cycles. Programming Mode: Dynamic - Refresh Related</p> |

DDRC_RFSHCTL3

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:7 | RO | 0x0 | reserved |
| 6:4 | RW | 0x0 | <p>refresh_mode Fine Granularity Refresh Mode 000: Fixed 1x (Normal mode) 001: Fixed 2x 010: Fixed 4x 101: Enable on the fly 2x (not supported) 110: Enable on the fly 4x (not supported) Everything else - reserved Note: Only Fixed 1x mode is supported if RFSHCTL3.dis_auto_refresh = 1. Note: The on-the-fly modes are not supported in this version of the uMCTL2. Note: This must be set up while the Controller is in reset or while the Controller is in self-refresh mode. Changing this during normal operation is not allowed. Making this a dynamic register will be supported in future version of the uMCTL2. Programming Mode: Quasi-dynamic Group 2</p> |
| 3:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | <p>refresh_update_level Toggle this signal (either from 0 to 1 or from 1 to 0) to indicate that the refresh register(s) have been updated. refresh_update_level must not be toggled when the DDRC is in reset (core_ddrc_rstn = 0). The refresh register(s) are automatically updated when exiting reset. Programming Mode: Dynamic</p> |
| 0 | RW | 0x0 | <p>dis_auto_refresh When '1', disable auto-refresh generated by the uMCTL2. When auto-refresh is disabled, the SoC core must generate refreshes using the registers DBGCMD.rankn_refresh. When dis_auto_refresh transitions from 0 to 1, any pending refreshes are immediately scheduled by the uMCTL2. Programming Mode: Dynamic - Refresh Related</p> |

DDRC RFSHTMG

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | t_rfc_nom_x1_sel Specifies whether the t_rfc_nom_x1_x32 register value is x1 or x32 |
| 30:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x062 | t_rfc_nom_x1_x32 tREFI: Average time interval between refreshes per rank (Specification: 7.8us for DDR2, DDR3. See JEDEC specification for mDDR, LPDDR2, LPDDR3). For LPDDR2/LPDDR3: if using all-bank refreshes (RFSHCTL0.per_bank_refresh= 0), this register should be set to tREFIab if using per-bank refreshes (RFSHCTL0.per_bank_refresh = 1), this register should be set to tREFIpb When the controller is operating in 1:2 frequency ratio mode, program this to (tREFI/2), no rounding up. In DDR mode, tREFI value is different depending on the refresh mode. The user should program the appropriate value from the spec based on the value programmed in the refresh mode register. Note that if RFSHTMG.t_rfc_nom_x1_sel==1, RFSHTMG.t_rfc_nom_x1_x32 must be greater than RFSHTMG.t_rfc_min; if RFSHTMG.t_rfc_nom_x1_sel==0, RFSHTMG.t_rfc_nom_x1_x32*32 must be greater than RFSHTMG.t_rfc_min; RFSHTMG.t_rfc_nom_x32 must be greater than 0x1. Unit: Clocks or multiples of 32 clocks, depending on RFSHTMG.t_rfc_nom_x1_sel. Programming Mode: Dynamic - Refresh Related |
| 15 | RW | 0x0 | lpddr3_trefbw_en Used only when LPDDR3 memory type is connected. Should only be changed when uMCTL2 is in reset. Specifies whether to use the tREFBW parameter (required by some LPDDR3 devices which comply with earlier versions of the LPDDR3 JEDEC specification) or not: 0: tREFBW parameter not used 1: tREFBW parameter used Programming Mode: Static |
| 14:10 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9:0 | RW | 0x08c | t_rfc_min tRFC (min): Minimum time from refresh to refresh or activate. t_rfc_min should be set to RoundUp(RoundUp(tRFCmin/tCK)/2). In LPDDR2/LPDDR3 mode: if using all-bank refreshes, the tRFCmin value in the above equations is equal to tRFCab if using per-bank refreshes, the tRFCmin value in the above equations is equal to tRFCpb Unit: DFI Clocks. Programming Mode: Dynamic - Refresh Related |

DDRC_CRCPARCTL0

Address: Operational Base + offset (0x00c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:3 | RO | 0x0 | reserved |
| 2 | W1C | 0x0 | dfi_alert_err_cnt_clr DFI alert error count clear. Clear bit for DFI alert error counter. Asserting this bit will clear the DFI alert error counter, CRCPARSTAT.dfi_alert_err_cnt. When the clear operation is complete, the uMCTL2 automatically clears this bit. Programming Mode: Dynamic |
| 1 | W1C | 0x0 | dfi_alert_err_int_clr Interrupt clear bit for DFI alert error. If this bit is set, the alert error interrupt on CRCPARSTAT.dfi_alert_err_int will be cleared. When the clear operation is complete, the uMCTL2 automatically clears this bit. Programming Mode: Dynamic |
| 0 | RW | 0x0 | dfi_alert_err_int_en Interrupt enable bit for DFI alert error. If this bit is set, any parity/CRC error detected on the dfi_alert_n input will result in an interrupt being set on CRCPARSTAT.dfi_alert_err_int. Programming Mode: Dynamic |

DDRC_CRCPARCTL1

Address: Operational Base + offset (0x00c4)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | <p>retry_add_rd_lat Retry additional read latency value. Delay value used is retry_add_rd_lat+1. Only used if CRCPARCTL1.retry_add_rd_lat_en is enabled.</p> <p>Selects the number of pipeline stages to dfi_rddata_valid/dfi_rddata/dfi_rddata_dbi before rest of internal uMCTL2 logic observes it.</p> <p>Required to compensate for fact delay in PHY/PCB for generating dfi_alert_n for retry may be more than the delay in PHY/PCB on read data path.</p> <p>Recommended settings (in terms of core_ddrc_core_clk): (Maximum Alert delay through PHY/PCB from erroneous read command including tPAR_UNKNOWN) - (Minimum Read data delay through PHY/PCB from erroneous read command) + (PHY's max granularity of dfi_rddata beats that may be corrupted before erroneous Read)</p> <p>Note: This calculation depends on various items such as RL, tPAR_ALERT_ON/tPAR_UNKNOWN/RCD/PHY/PCB behavior. Unit: DFI clock cycles. Programming Mode: Static</p> |
| 15 | RW | 0x0 | <p>retry_add_rd_lat_en Retry additional read latency enable. Number of pipeline stages selected is defined as CRCPARCTL1.retry_add_lat+1. Only set if CRCPARCTL1.crc_parity_retry_enable = 1</p> <p>Programming Mode: Static</p> |
| 14:13 | RO | 0x0 | reserved |
| 12 | RW | 0x1 | reserved |
| 11:8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | reserved |
| 6:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | <p>crc_enable CRC enable Register 1: Enable generation of CRC 0: Disable generation of CRC</p> <p>The setting of this register should match the CRC mode register setting in the DRAM.</p> <p>Programming Mode: Static</p> |
| 3:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 0 | RW | 0x0 | <p>parity_enable C/A Parity enable register 1: Enable generation of C/A parity and detection of C/A parity error 0: Disable generation of C/A parity and disable detection of C/A parity error If RCD's parity error detection or SDRAM's parity detection is enabled, this register should be 1. Programming Mode: Static</p> |

DDRC_CRCPARSTAT

Address: Operational Base + offset (0x00cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:18 | RO | 0x0 | reserved |
| 17 | RO | 0x0 | <p>dfi_alert_err_fatl_int Fatal parity error interrupt. One or more these situation below happens, this interrupt bit is set: MPSMX caused parity error. (RCD's parity error detection only) Parity error happens again during software intervention time MRS was in retry_fifo_max_hold_timer_x4 window from alert_n=0 or STAT.operating_mode is Init. It remains set until cleared by CRCPARCTL0.dfi_alert_err_fatl_clr. If this interrupt is asserted, system reset is strongly recommended. Programming Mode: Static</p> |
| 16 | RO | 0x0 | <p>dfi_alert_err_int DFI alert error interrupt. If a parity/CRC error is detected on dfi_alert_n, and the interrupt is enabled by CRCPARCTL0.dfi_alert_err_int_en, this interrupt bit will be set. It will remain set until cleared by CRCPARCTL0.dfi_alert_err_int_clr Programming Mode: Static</p> |
| 15:0 | RO | 0x0000 | <p>dfi_alert_err_cnt DFI alert error count. If a parity/CRC error is detected on dfi_alert_n, this counter be incremented. This is independent of the setting of CRCPARCTL0.dfi_alert_err_int_en. It will saturate at 0xFFFF, and can be cleared by asserting CRCPARCTL0.dfi_alert_err_cnt_clr. Programming Mode: Static</p> |

DDRC_INITO

Address: Operational Base + offset (0x00d0)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:30 | RW | 0x0 | <p>skip_dram_init If lower bit is enabled the SDRAM initialization routine is skipped. The upper bit decides what state the controller starts up in when reset is removed</p> <p>00: SDRAM Initialization routine is run after power-up 01: SDRAM Initialization routine is skipped after powerup. Controller starts up in Normal Mode 11: SDRAM Initialization routine is skipped after powerup. Controller starts up in Self-refresh Mode 10: SDRAM Initialization routine is run after power-up. Programming Mode: Quasi-dynamic Group 2</p> |
| 29:26 | RO | 0x0 | reserved |
| 25:16 | RW | 0x002 | <p>post_cke_x1024 Cycles to wait after driving CKE high to start the SDRAM initialization sequence. Unit: 1024 DFI clock cycles. DDR2 typically requires a 400 ns delay, requiring this value to be programmed to 2 at all clock speeds. LPDDR2/LPDDR3 typically requires this to be programmed for a delay of 200 us. When the controller is operating in 1:2 frequency ratio mode, program this to JEDEC spec value divided by 2, and round it up to the next integer value. Programming Mode: Static</p> |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x04e | <p>pre_cke_x1024 Cycles to wait after reset before driving CKE high to start the SDRAM initialization sequence. Unit: 1024 DFI clock cycles. DDR2 specifications typically require this to be programmed for a delay of \geq 200 us. LPDDR2/LPDDR3: tINIT1 of 100 ns (min) When the controller is operating in 1:2 frequency ratio mode, program this to JEDEC spec value divided by 2, and round it up to the next integer value. Programming Mode: Static</p> |

DDRC_INIT1

Address: Operational Base + offset (0x00d4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | reserved |
| 24:16 | RW | 0x000 | dram_rstn_x1024 Number of cycles to assert SDRAM reset signal during init sequence. This is only present for designs supporting DDR3, DDR devices. When the controller is operating in 1:2 frequency ratio mode, program this to JEDEC spec value divided by 2, and round it up to the next integer value. Unit: 1024 DFI clock cycles. Programming Mode: Static |
| 15:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | pre_ocd_x32 Wait period before driving the OCD complete command to SDRAM. Unit: Counts of a global timer that pulses every 32 DFI clock cycles. There is no known specific requirement for this; it may be set to zero. Programming Mode: Static |

DDRC INIT2

Address: Operational Base + offset (0x00d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15:8 | RW | 0x0d | idle_after_reset_x32 Idle time after the reset command, tINIT4. Present only in designs configured to support LPDDR2. When the controller is operating in 1:2 frequency ratio mode, program this to JEDEC spec value divided by 2, and round it up to the next integer value. Unit: 32 DFI clock cycles. Programming Mode: Static |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x5 | min_stable_clock_x1 Time to wait after the first CKE high, tINIT2. Present only in designs configured to support LPDDR2/LPDDR3. LPDDR2/LPDDR3 typically requires 5 x tCK delay. When the controller is operating in 1:2 frequency ratio mode, program this to JEDEC spec value divided by 2, and round it up to the next integer value. Unit: DFI clock cycles. Programming Mode: Static |

DDRC INIT3

Address: Operational Base + offset (0x00dc)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RW | 0x0000 | <p>mr</p> <p>DDR2: Value to write to MR register. Bit 8 is for DLL and the setting here is ignored. The uMCTL2 sets this bit appropriately.</p> <p>DDR3: Value loaded into MR0 register.</p> <p>mDDR: Value to write to MR register.</p> <p>LPDDR2/LPDDR3 - Value to write to MR1 register</p> <p>Programming Mode: Quasi-dynamic Group 1 and Group 4</p> |
| 15:0 | RW | 0x0510 | <p>emr</p> <p>DDR2: Value to write to EMR register. Bits 9:7 are for OCD and the setting in this register is ignored. The uMCTL2 sets those bits appropriately.</p> <p>DDR3: Value to write to MR1 register Set bit 7 to 0. If PHY-evaluation mode training is enabled, this bit is set appropriately by the uMCTL2 during write leveling.</p> <p>mDDR: Value to write to EMR register.</p> <p>LPDDR2/LPDDR3 - Value to write to MR2 register</p> <p>Programming Mode: Quasi-dynamic Group 4</p> |

DDRC INIT4

Address: Operational Base + offset (0x00e0)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | <p>emr2</p> <p>DDR2: Value to write to EMR2 register.</p> <p>DDR3/DDR: Value to write to MR2 register</p> <p>LPDDR2/LPDDR3: Value to write to MR3 register</p> <p>mDDR: Unused</p> <p>Programming Mode: Quasi-dynamic Group 4</p> |
| 15:0 | RW | 0x0000 | <p>emr3</p> <p>DDR2: Value to write to EMR3 register.</p> <p>DDR3: Value to write to MR3 register</p> <p>mDDR/LPDDR2/LPDDR3: Unused</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

DDRC INIT5

Address: Operational Base + offset (0x00e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x10 | <p>dev_zqinit_x32 ZQ initial calibration, tZQINIT. Present only in designs configured to support DDR3 or LPDDR2/LPDDR3. DDR3 typically requires 512 SDRAM clock cycles. LPDDR2/LPDDR3 requires 1 us.</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to JEDEC spec value divided by 2, and round it up to the next integer value.</p> <p>Unit: 32 DFI clock cycles. Programming Mode: Static</p> |
| 15:10 | RO | 0x0 | reserved |
| 9:0 | RW | 0x004 | <p>max_auto_init_x1024 Maximum duration of the auto initialization, tINIT5. Present only in designs configured to support LPDDR2/LPDDR3. LPDDR2/LPDDR3 typically requires 10 us.</p> <p>Unit: 1024 DFI clock cycles. Programming Mode: Static</p> |

DDRC INIT6

Address: Operational Base + offset (0x00e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:16 | RW | 0x0000 | reserved |
| 15:0 | RW | 0x0000 | reserved |

DDRC INIT7

Address: Operational Base + offset (0x00ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:16 | RW | 0x0000 | reserved |
| 15:0 | RW | 0x0000 | reserved |

DDRC DIMMCTL

Address: Operational Base + offset (0x00f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:5 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 4 | RW | 0x0 | <p>mrs_bg1_en Enable for BG1 bit of MRS command. BG1 bit of the mode register address is specified as RFU (Reserved for Future Use) and must be programmed to 0 during MRS. In case where DRAMs which do not have BG1 are attached and both the CA parity and the Output Inversion are enabled, this must be set to 0, so that the calculation of CA parity will not include BG1 bit.</p> <p>Note: This has no effect on the address of any other memory accesses, or of software-driven mode register accesses. If address mirroring is enabled, this is applied to BG1 of even ranks and BG0 of odd ranks.</p> <p>1: Enabled; 0: Disabled Programming Mode: Static</p> |
| 3 | RW | 0x0 | <p>mrs_a17_en Enable for A17 bit of MRS command. A17 bit of the mode register address is specified as RFU (Reserved for Future Use) and must be programmed to 0 during MRS. In case where DRAMs which do not have A17 are attached and the Output Inversion are enabled, this must be set to 0, so that the calculation of CA parity will not include A17 bit.</p> <p>Note: This has no effect on the address of any other memory accesses, or of software-driven mode register accesses.</p> <p>1: Enabled; 0: Disabled Programming Mode: Static</p> |
| 2:0 | RO | 0x0 | reserved |

DDRC_RANKCTL

Address: Operational Base + offset (0x00f4)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x6 | <p>diff_rank_wr_gap Only present for multi-rank configurations. Indicates the number of clocks of gap in data responses when performing consecutive writes to different ranks. This is used to switch the delays in the PHY to match the rank requirements. This value should consider both PHY requirement and ODT requirement.</p> <p>PHY requirement: tphy_wrcsgap (see PHY databook for value of tphy_wrcsgap) If CRC feature is enabled, should be increased by 1.</p> <p>Programming Mode: Quasi-dynamic Group 2</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 7:4 | RW | 0x6 | <p>diff_rank_rd_gap Only present for multi-rank configurations. Indicates the number of clocks of gap in data responses when performing consecutive reads to different ranks. This is used to switch the delays in the PHY to match the rank requirements. This value should consider both PHY requirement and ODT requirement.</p> <p>PHY requirement: tphy_rdcsgap (see PHY databook for value of tphy_rdcsgap)</p> <p>ODT requirement: The value programmed in this register takes care of the ODT switch off timing requirement when switching ranks during reads. When the controller is operating in 1:2 mode, program this to the larger value divided by two and round it up to the next integer.</p> <p>Programming Mode: Static</p> |
| 3:0 | RW | 0xf | <p>max_rank_rd Only present for multi-rank configurations.</p> <p>Background: Reads to the same rank can be performed back-to-back. Reads to different ranks require additional gap dictated by the register RANKCTL.diff_rank_rd_gap. This is to avoid possible data bus contention as well as to give PHY enough time to switch the delay when changing ranks. The uMCTL2 arbitrates for bus access on a cycle-by-cycle basis; therefore after a read is scheduled, there are few clock cycles (determined by the value on RANKCTL.diff_rank_rd_gap register) in which only reads from the same rank are eligible to be scheduled. This prevents reads from other ranks from having fair access to the data bus.</p> <p>This parameter represents the maximum number of reads that can be scheduled consecutively to the same rank. After this number is reached, a delay equal to RANKCTL.diff_rank_rd_gap is inserted by the scheduler to allow all ranks a fair opportunity to be scheduled. Higher numbers increase bandwidth utilization, lower numbers increase fairness.</p> <p>This feature can be DISABLED by setting this register to 0. When set to 0, the Controller will stay on the same rank as long as commands are available for it.</p> <p>Minimum programmable value is 0 (feature disabled) and maximum programmable value is 0xF.</p> <p>Programming Mode: Static</p> |

DDRC_DRAMTMG0

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|-------------|
| 31 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 30:24 | RW | 0x0f | <p>wr2pre Minimum time between write and precharge to same bank. Unit: DFI Clocks Specifications: WL + BL/2 + tWR = approximately 8 cycles + 15 ns = 14 clocks @400MHz and less for lower frequencies where: WL = write latency BL = burst length. This must match the value programmed in the BL bit of the mode register to the SDRAM. BST (burst terminate) is not supported at present. tWR = Write recovery time. This comes directly from the SDRAM specification. Add one extra cycle for LPDDR2/LPDDR3 for this parameter. When the controller is operating in 1:2 frequency ratio mode, 1T mode, divide the above value by 2. No rounding up. When the controller is operating in 1:2 frequency ratio mode, 2T mode mode, divide the above value by 2 and round it up to the next integer value. Programming Mode: Quasi-dynamic Group 1 and Group 2 and Group 4</p> |
| 23:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x10 | <p>t_faw tFAW Valid only when 8 or more banks(or banks x bank groups) are present. In 8-bank design, at most 4 banks must be activated in a rolling window of tFAW cycles. When the controller is operating in 1:2 frequency ratio mode, program this to (tFAW/2) and round up to next integer value. In a 4-bank design, set this register to 0x1 independent of the 1:1/1:2 frequency mode. Unit: DFI Clocks Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 15 | RO | 0x0 | reserved |
| 14:8 | RW | 0x1b | <p>t_ras_max tRAS(max): Maximum time between activate and precharge to same bank. This is the maximum time that a page can be kept open Minimum value of this register is 1. Zero is invalid. When the controller is operating in 1:2 frequency ratio mode, program this to (tRAS(max)-1)/2. No rounding up. Unit: Multiples of 1024 DFI clocks. Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 7:6 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5:0 | RW | 0x0f | t_ras_min tRAS(min): Minimum time between activate and precharge to the same bank. When the controller is operating in 1:2 frequency mode, 1T mode, program this to tRAS(min)/2. No rounding up. When the controller is operating in 1:2 frequency ratio mode, 2T mode mode, program this to (tRAS(min)/2) and round it up to the next integer value. Unit: DFI Clocks Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC DRAMTMG1

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x08 | t_xp tXP: Minimum time after power-down exit to any operation. For DDR3, this should be programmed to tXPDLL if slow powerdown exit is selected in MR0[12]. When the controller is operating in 1:2 frequency ratio mode, program this to (tXP/2) and round it up to the next integer value. Units: DFI Clocks Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x04 | rd2pre tRTP: Minimum time from read to precharge of same bank. DDR2: tAL + BL/2 + max(tRTP, 2) - 2 DDR3: tAL + max (tRTP, 4) mDDR: BL/2 LPDDR2: Depends on if it's LPDDR2-S2 or LPDDR2-S4: LPDDR2-S2: BL/2 + tRTP - 1. LPDDR2-S4: BL/2 + max(tRTP,2) - 2. LPDDR3: BL/2 + max(tRTP,4) - 4 When the controller is operating in 1:2 mode, 1T mode, divide the above value by 2. No rounding up. When the controller is operating in 1:2 mode, 2T mode, divide the above value by 2 and round it up to the next integer value. Unit: DFI Clocks. Programming Mode: Quasi-dynamic Group 1 and Group 2 and Group 4 |
| 7 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 6:0 | RW | 0x14 | t_rc tRC: Minimum time between activates to same bank. When the controller is operating in 1:2 frequency ratio mode, program this to (tRC/2) and round up to next integer value. Unit: DFI Clocks. Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC_DRAMTMG2

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:24 | RW | 0x03 | write_latency Set to WL. Time from write command to write data on SDRAM interface. For mDDR, it should normally be set to 1. When the controller is operating in 1:2 frequency ratio mode, divide the value calculated using the above equation by 2, and round it up to next integer. This register field is not required for DDR2 and DDR3 (except if MEMC_TRAINING is set), as the DFI read and write latencies defined in DFITMG0 and DFITMG1 are sufficient for those protocols Unit: DFI clocks Programming Mode: Quasi-dynamic Group 1 and Group 2 and Group 4 |
| 23:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x05 | read_latency Set to RL. Time from read command to read data on SDRAM interface. When the controller is operating in 1:2 frequency ratio mode, divide the value calculated using the above equation by 2, and round it up to next integer. This register field is not required for DDR2 and DDR3 (except if MEMC_TRAINING is set), as the DFI read and write latencies defined in DFITMG0 and DFITMG1 are sufficient for those protocols Unit: DFI clocks Programming Mode: Quasi-dynamic Group 1 and Group 2 and Group 4 |
| 15:14 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 13:8 | RW | 0x06 | <p>rd2wr DDR2/3/mDDR: RL + BL/2 + 2 - WL LPDDR2/LPDDR3: RL + BL/2 + RU(tDQSCKmax/tCK) + 1 - WL Minimum time from read command to write command. Include time for bus turnaround and all per-bank, per-rank, and global constraints. Please see the relevant PHY databook for details of what should be included here. Unit: DFI Clocks. Where: WL = write latency BL = burst length. This must match the value programmed in the BL bit of the mode register to the SDRAM RL = read latency = CAS latency RD_POSTAMBLE = read postamble. This is unique to LPDDR. For LPDDR2/LPDDR3, if derating is enabled (DERATEEN.derate_enable=1), derated tDQSCKmax should be used. When the controller is operating in 1:2 frequency ratio mode, divide the value calculated using the above equation by 2, and round it up to next integer. Programming Mode: Quasi-dynamic Group 1 and Group 2 and Group 4</p> |
| 7:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x0d | <p>wr2rd LPDDR2/3/4: WL + BL/2 + tWTR + 1 Others: CWL + BL/2 + tWTR Unit: DFI Clocks. Where: CWL = CAS write latency WL = Write latency PL = Parity latency BL = burst length. This must match the value programmed in the BL bit of the mode register to the SDRAM tWTR_L = internal write to read command delay for same bank group. This comes directly from the SDRAM specification. tWTR = internal write to read command delay. This comes directly from the SDRAM specification. Add one extra cycle for LPDDR2/LPDDR3 operation. When the controller is operating in 1:2 mode, divide the value calculated using the above equation by 2, and round it up to next integer. Programming Mode: Quasi-dynamic Group 1 and Group 2 and Group 4</p> |

DDRC_DRAMTMG3

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29:20 | RW | 0x005 | <p>t_mrw Time to wait after a mode register write or read (MRW or MRR). Present only in designs configured to support LPDDR2, LPDDR3. LPDDR2 typically requires value of 5. LPDDR3 typically requires value of 10.</p> <p>For LPDDR2, this register is used for the time from a MRW/MRR to all other commands.</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to the above values divided by 2 and round it up to the next integer value.</p> <p>For LPDDR3, this register is used for the time from a MRW/MRR to a MRW/MRR.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 19:18 | RO | 0x0 | reserved |
| 17:12 | RW | 0x04 | <p>t_mrd tMRD: Cycles to wait after a mode register write or read. Depending on the connected SDRAM, tMRD represents: DDR2/mDDR: Time from MRS to any command DDR3: Time from MRS to MRS command LPDDR2: not used LPDDR3: Time from MRS to non-MRS command.</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to (tMRD/2) and round it up to the next integer value.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 11:10 | RO | 0x0 | reserved |
| 9:0 | RW | 0x00c | <p>t_mod tMOD: Parameter used only in DDR3 and DDR. Cycles between load mode command and following non-load mode command. Set to tMOD/2 (rounded up to next integer) if controller is operating in 1:2 frequency ratio mode.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

DDRC_DRAMTMG4

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:29 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 28:24 | RW | 0x05 | t_rcd tRCD - tAL: Minimum time from activate to read or write command to same bank. When the controller is operating in 1:2 frequency ratio mode, program this to ((tRCD - tAL)/2) and round it up to the next integer value. Minimum value allowed for this register is 1, which implies minimum (tRCD - tAL) value to be 2 when the controller is operating in 1:2 frequency ratio mode. Unit: DFI Clocks. Programming Mode: Quasi-dynamic Group 1 and Group 2 and Group 4 |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RW | 0x4 | t_ccd Others: tCCD: This is the minimum time between two reads or two writes. When the controller is operating in 1:2 frequency ratio mode, program this to (tCCD_L/2 or tCCD/2) and round it up to the next integer value. Unit: DFI clocks. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 15:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x4 | t_rrd Others: tRRD: Minimum time between activates from bank "a" to bank "b" When the controller is operating in 1:2 frequency ratio mode, program this to (tRRD_L/2 or tRRD/2) and round it up to the next integer value. Unit: DFI Clocks. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x05 | t_rp tRP: Minimum time from precharge to activate of same bank. When the controller is operating in 1:2 frequency ratio mode, t_rp should be set to RoundDown(RoundUp(tRP/tCK)/2) + 1. Unit: DFI Clocks. Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC DRAMTMG5

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|-------------|
| 31:28 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 27:24 | RW | 0x5 | <p>t_cksrx This is the time before Self Refresh Exit that CK is maintained as a valid clock before issuing SRX. Specifies the clock stable time before SRX.</p> <p>Recommended settings: mDDR: 1 LPDDR2: 2 LPDDR3: 2 DDR2: 1 DDR3: tCKSRX</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RW | 0x5 | <p>t_cksrre This is the time after Self Refresh Down Entry that CK is maintained as a valid clock. Specifies the clock disable delay after SRE.</p> <p>Recommended settings: mDDR: 0 LPDDR2: 2 LPDDR3: 2 DDR2: 1 DDR3: max (10 ns, 5 tCK)</p> <p>(*)Only if CRCPARCTL1.caparity_disable_before_sr=0, this register should be increased by PL.</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 15:14 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 13:8 | RW | 0x04 | <p>t_ckesr Minimum CKE low width for Self refresh or Self refresh power down entry to exit timing in memory clock cycles. Recommended settings: mDDR: tRFC LPDDR2: tCKESR LPDDR3: tCKESR</p> <p>DDR2: tCKE DDR3: tCKE + 1 (*)Only if CRCPARCTL1.capacity_disable_before_sr=0, this register should be increased by PL. When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer. Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x03 | <p>t_cke Minimum number of cycles of CKE HIGH/LOW during power-down and self refresh. LPDDR2/LPDDR3 mode: Set this to the larger of tCKE or tCKESR Non-LPDDR2/non-LPDDR3 designs: Set this to tCKE value. When the controller is operating in 1:2 frequency ratio mode, program this to (value described above)/2 and round it up to the next integer value. Unit: DFI Clocks. Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

DDRC_DRAMTMG6

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:24 | RW | 0x2 | <p>t_ckdpde This is the time before Self Refresh Exit that CK is maintained as a valid clock before issuing SRX. Specifies the clock stable time before SRX. Recommended settings: mDDR: 1 LPDDR2: 2 LPDDR3: 2 DDR2: 1 DDR3: tCKSRX When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer. Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RW | 0x2 | <p>t_ckdpdx</p> <p>This is the time before Deep Power Down Exit that CK is maintained as a valid clock before issuing DPDX. Specifies the clock stable time before DPDX.</p> <p>Recommended settings:</p> <p>mDDR: 1 LPDDR2: 2 LPDDR3: 2</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer.</p> <p>This is only present for designs supporting mDDR or LPDDR2 devices.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 15:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x5 | <p>t_ckcsx</p> <p>This is the time before Clock Stop Exit that CK is maintained as a valid clock before issuing Clock Stop Exit. Specifies the clock stable time before next command after Clock Stop Exit.</p> <p>Recommended settings:</p> <p>mDDR: 1 LPDDR2: tXP + 2 LPDDR3: tXP + 2</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

DDRC_DRAMTMG7

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:8 | RW | 0x2 | <p>t_ckpde This is the time after Power Down Entry that CK is maintained as a valid clock. Specifies the clock disable delay after PDE. Recommended settings: mDDR: 0 LPDDR2: 2 LPDDR3: 2 When using DDR2/3/4 SDRAM, this register should be set to the same value as DRAMTMG5.t_cksre. When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer. Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x2 | <p>t_ckpdx This is the time before Power Down Exit that CK is maintained as a valid clock before issuing PDX. Specifies the clock stable time before PDX. Recommended settings: mDDR: 0 LPDDR2: 2 LPDDR3: 2 When using DDR2/3/4 SDRAM, this register should be set to the same value as DRAMTMG5.t_cksrx. When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer. This is only present for designs supporting mDDR or LPDDR2/LPDDR3 devices. Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

DDRC_DRAMTMG8

Address: Operational Base + offset (0x0120)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30:24 | RW | 0x03 | <p>t_xs_fast_x32 tXS_FAST: Exit Self Refresh to ZQCL, ZQCS and MRS (only CL, WR, RTP and Geardown mode). When the controller is operating in 1:2 frequency ratio mode, program this to the above value divided by 2 and round up to next integer value. Unit: Multiples of 32 DFI clocks. Note: This is applicable to only ZQCL/ZQCS commands. Note: Ensure this is less than or equal to t_xs_x32. Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 23 | RO | 0x0 | reserved |
| 22:16 | RW | 0x03 | t_xs_abort_x32 tXS_ABORT: Exit Self Refresh to commands not requiring a locked DLL in Self Refresh Abort. When the controller is operating in 1:2 frequency ratio mode, program this to the above value divided by 2 and round up to next integer value. Unit: Multiples of 32 DFI clocks. Note: Ensure this is less than or equal to t_xs_x32. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 15 | RO | 0x0 | reserved |
| 14:8 | RW | 0x44 | t_xs_dll_x32 tXSDLL: Exit Self Refresh to commands requiring a locked DLL. When the controller is operating in 1:2 frequency ratio mode, program this to the above value divided by 2 and round up to next integer value. Unit: Multiples of 32 DFI clocks. Note: Used only for DDR2, DDR3 SDRAMs. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 7 | RO | 0x0 | reserved |
| 6:0 | RW | 0x05 | t_xs_x32 tXS: Exit Self Refresh to commands not requiring a locked DLL. When the controller is operating in 1:2 frequency ratio mode, program this to the above value divided by 2 and round up to next integer value. Unit: Multiples of 32 DFI clocks. Note: Used only for DDR2, DDR3 SDRAMs. Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC_DRAMTMG9

Address: Operational Base + offset (0x0124)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31 | RO | 0x0 | reserved |
| 30 | RW | 0x0 | reserved |
| 29:19 | RO | 0x0 | reserved |
| 18:16 | RW | 0x4 | reserved |
| 15:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x4 | reserved |
| 7:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x0d | reserved |

DDRC_DRAMTMG10

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x1c | t_sync_gear Indicates the time between MRS command and the sync pulse time. This must be even number of clocks. tMOD(min) is greater of 24nCK or 15ns $15\text{ns} / .625\text{ns} = 24$ Max value for this register is $24+4 = 28$ When the controller is operating in 1:2 mode, program this to (tSYNC_GEAR/2) and round it up to the next integer value. Unit: DFI Clocks Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x18 | t_cmd_gear Sync pulse to first valid command. tMOD(min) is greater of 24nCK or 15ns $15\text{ns} / .625\text{ns} = 24$ Max value for this register is 24 When the controller is operating in 1:2 mode, program this to (tCMD_GEAR/2) and round it up to the next integer value. Unit: DFI Clocks Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 7:4 | RO | 0x0 | reserved |
| 3:2 | RW | 0x2 | t_gear_setup Geardown setup time. Minimum value of this register is 1. Zero is invalid. When the controller is operating in 1:2 frequency ratio mode, program this to (tGEAR_setup/2) and round it up to the next integer value. Unit: DFI Clocks Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 1:0 | RW | 0x2 | t_gear_hold Geardown hold time. Minimum value of this register is 1. Zero is invalid. When the controller is operating in 1:2 frequency ratio mode, program this to (tGEAR_hold/2) and round it up to the next integer value. Unit: DFI Clocks Value After Reset: 0x2 Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC_DRAMTMG11

Address: Operational Base + offset (0x012c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 30:24 | RW | 0x44 | reserved |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x0c | t_mpx_lh tMPX_LH: This is the minimum CS_n Low hold time to CKE rising edge. When the controller is operating in 1:2 frequency ratio mode, program this to RoundUp(tMPX_LH/2)+1. Present only in designs configured to support DDR. Unit: DFI clocks. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 15:10 | RO | 0x0 | reserved |
| 9:8 | RW | 0x2 | reserved |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x1c | reserved |

DDRC_DRAMTMG12

Address: Operational Base + offset (0x0130)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:18 | RO | 0x0 | reserved |
| 17:16 | RW | 0x2 | t_cmdcke tCMDCKE: Delay from valid command to CKE input LOW. Set this to the larger of tESCKE or tCMDCKE When the controller is operating in 1:2 frequency ratio mode, program this to (max(tESCKE, tCMDCKE)/2) and round it up to the next integer value. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 15:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x10 | t_mrd_pda tMRD_PDA: This is the Mode Register Set command cycle time in PDA mode. When the controller is operating in 1:2 frequency ratio mode, program this to (tMRD_PDA/2) and round it up to the next integer value. Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC_DRAMTMG13

Address: Operational Base + offset (0x0134)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31 | RO | 0x0 | reserved |
| 30:24 | RW | 0x1c | reserved |
| 23:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x20 | reserved |
| 15:3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x4 | reserved |

DDRC DRAMTMG14

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x0a0 | t_xsr tXSR: Exit Self Refresh to any command. When the controller is operating in 1:2 frequency ratio mode, program this to the above value divided by 2 and round up to next integer value. Note: Used only for mDDR/LPDDR2/LPDDR3 mode. Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC DRAMTMG15

Address: Operational Base + offset (0x013c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | en_dfi_lp_t_stab 1: Enable using tSTAB when exiting DFI LP. Needs to be set when the PHY is stopping the clock during DFI LP to save maximum power. 0: Disable using tSTAB when exiting DFI LP Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 30:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | t_stab_x32 tSTAB: Stabilization time. It is required in the following two cases for DDR3 RDIMM : when exiting power saving mode, if the clock was stopped, after re-enabling it the clock must be stable for a time specified by tSTAB in the case of input clock frequency change after issuing control words that refers to clock timing (Specification: 6us for DDR3) When the controller is operating in 1:2 frequency ratio mode, program this to recommended value divided by two and round it up to next integer. Unit: Multiples of 32 DFI clock cycles. Programming Mode: Quasi-dynamic Group 2 and Group 4 |

DDRC DRAMTMG17

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x00 | t_vrcg_enable When the controller is operating in 1:2 frequency ratio mode, program this to (tVRCG_ENABLE/2) and round it up to the next integer value. Unit: DFI clocks Programming Mode: Quasi-dynamic Group 4 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 15:7 | RO | 0x0 | reserved |
| 6:0 | RW | 0x00 | t_vrcg_disable When the controller is operating in 1:2 frequency ratio mode, program this to (tVRCG_DISABLE/2) and round it up to the next integer value. Unit: DFI clocks Programming Mode: Quasi-dynamic Group 4 |

DDRC_ZQCTL0

Address: Operational Base + offset (0x0180)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | dis_auto_zq 1: Disable uMCTL2 generation of ZQCS/MPC(ZQ calibration) command. Register DBGCMD.zq_calib_short can be used instead to issue ZQ calibration request from APB module. 0: Internally generate ZQCS/MPC(ZQ calibration) commands based on ZQCTL1.t_zq_short_interval_x1024. This is only present for designs supporting DDR3 or LPDDR2/LPDDR3 devices. Programming Mode: Dynamic |
| 30 | RW | 0x0 | dis_srx_zql 1: Disable issuing of ZQCL/MPC(ZQ calibration) command at Self-Refresh/SR-Powerdown exit. Only applicable when run in DDR3 or LPDDR2 or LPDDR3 mode. 0: Enable issuing of ZQCL/MPC(ZQ calibration) command at Self-Refresh/SR-Powerdown exit. Only applicable when run in DDR3 or LPDDR2 or LPDDR3 mode. This is only present for designs supporting DDR3/DDR or LPDDR2/LPDDR3 devices. Programming Mode: Quasi-dynamic Group 2 and Group 4 |
| 29 | RW | 0x0 | zq_resistor_shared 1: Denotes that ZQ resistor is shared between ranks. Means ZQinit/ZQCL/ZQCS/MPC(ZQ calibration) commands are sent to one rank at a time with tZQinit/tZQCL/tZQCS/tZQCAL/tZQLAT timing met between commands so that commands to different ranks do not overlap. 0: ZQ resistor is not shared. This is only present for designs supporting DDR3 or LPDDR2/LPDDR3 devices. Programming Mode: Static |
| 28 | RW | 0x0 | reserved |
| 27 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 26:16 | RW | 0x200 | t_zq_long_nop tZQoper for DDR3, tZQCL for LPDDR2/LPDDR3, Number of DFI clock cycles of NOP required after a ZQCL (ZQ calibration long)/MPC(ZQ Start) command is issued to SDRAM. When the controller is operating in 1:2 frequency ratio mode: DDR3: program this to tZQoper/2 and round it up to the next integer value. LPDDR2/LPDDR3: program this to tZQCL/2 and round it up to the next integer value. This is only present for designs supporting DDR3/DDR or LPDDR2/LPDDR3/LPDDR devices. Programming Mode: Static |
| 15:10 | RO | 0x0 | reserved |
| 9:0 | RW | 0x040 | t_zq_short_nop tZQCS for DDR3/DD4/LPDDR2/LPDDR3, tzQLAT for LPDDR: Number of DFI clock cycles of NOP required after a ZQCS (ZQ calibration short)/MPC(ZQ Latch) command is issued to SDRAM. When the controller is operating in 1:2 frequency ratio mode, program this to tZQCS/2 and round it up to the next integer value. This is only present for designs supporting DDR3/DDR or LPDDR2/LPDDR3/LPDDR devices. Programming Mode: Static |

DDRC_ZQCTL1

Address: Operational Base + offset (0x0184)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:20 | RW | 0x020 | t_zq_reset_nop tZQReset: Number of DFI clock cycles of NOP required after a ZQReset (ZQ calibration Reset) command is issued to SDRAM. When the controller is operating in 1:2 frequency ratio mode, program this to tZQReset/2 and round it up to the next integer value. This is only present for designs supporting LPDDR2/LPDDR3/LPDDR devices. Programming Mode: Static |
| 19:0 | RW | 0x00100 | t_zq_short_interval_x1024 Average interval to wait between automatically issuing ZQCS (ZQ calibration short)/MPC(ZQ calibration) commands to DDR3/DDR/LPDDR2/LPDDR3/LPDDR devices. Meaningless, if ZQCTL0.dis_auto_zq=1. Unit: 1024 DFI clock cycles. This is only present for designs supporting DDR3/DDR or LPDDR2/LPDDR3/LPDDR devices. Programming Mode: Static |

DDRC_ZQCTL2

Address: Operational Base + offset (0x0188)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | zq_reset Setting this register bit to 1 triggers a ZQ Reset operation. When the ZQ Reset operation is complete, the uMCTL2 automatically clears this bit. It is recommended NOT to set this signal if in Init, Self-Refresh(except LPDDR) or SRPowerdown(LPDDR) or Deep power-down operating modes. This is only present for designs supporting LPDDR2/LPDDR3/LPDDR devices. Programming Mode: Dynamic |

DDRC_ZQSTAT

Address: Operational Base + offset (0x018c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RW | 0x0 | <p>zq_reset_busy SoC core may initiate a ZQ Reset operation only if this signal is low. This signal goes high in the clock after the uMCTL2 accepts the ZQ Reset request. It goes low when the ZQ Reset command is issued to the SDRAM and the associated NOP period is over. It is recommended not to perform ZQ Reset commands when this signal is high.</p> <p>0: Indicates that the SoC core can initiate a ZQ Reset operation 1: Indicates that ZQ Reset operation is in progress Programming Mode: Dynamic</p> |

DDRC DFITMG0

Address: Operational Base + offset (0x0190)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:29 | RO | 0x0 | reserved |
| 28:24 | RW | 0x07 | <p>dfi_t_ctrl_delay Specifies the number of DFI clock cycles after an assertion or de-assertion of the DFI control signals that the control signals at the PHY-DRAM interface reflect the assertion or de-assertion. If the DFI clock and the memory clock are not phase-aligned, this timing parameter should be rounded up to the next integer value. Programming Mode: Quasi-dynamic Group 4</p> |
| 23 | RW | 0x0 | <p>dfi_rddata_use_dfi_phy_clk Defines whether dfi_rddata_en/dfi_rddata/dfi_rddata_valid is generated using HDR (DFI clock) or SDR (DFI PHY clock) values. Selects whether value in DFITMG0.dfi_t_rddata_en is in terms of HDR (DFI clock) or SDR (DFI PHY clock) cycles: 0: in terms of HDR (DFI clock) cycles, Only support HDR DFI clock. 1: in terms of SDR (DFI PHY clock) cycles Programming Mode: Static</p> |
| 22:16 | RW | 0x02 | <p>dfi_t_rddata_en Time from the assertion of a read command on the DFI interface to the assertion of the dfi_rddata_en signal. Refer to PHY specification for correct value. This corresponds to the DFI parameter trddata_en. Unit: DFI clock cycles Programming Mode: Quasi-dynamic Group 1 and Group 4</p> |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 15 | RW | 0x0 | <p>dfi_wrdata_use_dfi_phy_clk Defines whether dfi_wrdata_en=dfi_wrdata=dfi_wrdata_mask is generated using HDR (DFI clock) or SDR (DFI PHY clock) values Selects whether value in DFITMG0.dfi_tphy_wrlat is in terms of HDR (DFI clock) or SDR (DFI PHY clock) cycles Selects whether value in DFITMG0.dfi_tphy_wrdata is in terms of HDR (DFI clock) or SDR (DFI PHY clock) cycles 0: in terms of HDR (DFI clock) cycles, Only support HDR DFI clock. 1: in terms of SDR (DFI PHY clock) cycles Programming Mode: Static</p> |
| 14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | <p>dfi_tphy_wrdata Specifies the number of clock cycles between when dfi_wrdata_en is asserted to when the associated write data is driven on the dfi_wrdata signal. This corresponds to the DFI timing parameter tphy_wrdata. Refer to PHY specification for correct value. Note, max supported value is 8. Unit: DFI clock cycles. Programming Mode: Quasi-dynamic Group 4</p> |
| 7:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x02 | <p>dfi_tphy_wrlat Write latency Number of clocks from the write command to write data enable (dfi_wrdata_en). This corresponds to the DFI timing parameter tphy_wrlat. Refer to PHY specification for correct value. For LPDDR, dfi_tphy_wrlat>60 is not supported. Unit: DFI clock cycles Programming Mode: Quasi-dynamic Group 1 and Group 4</p> |

DDRC DFITMG1

Address: Operational Base + offset (0x0194)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:28 | RW | 0x0 | <p>dfi_t_cmd_lat Specifies the number of DFI PHY clock cycles between when the dfi_cs signal is asserted and when the associated command is driven. This field is used for CAL mode, should be set to '0' or the value which matches the CAL mode register setting in the DRAM. If the PHY can add the latency for CAL mode, this should be set to '0'. Valid Range: 0, 3, 4, 5, 6, and 8 Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 27:26 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 25:24 | RW | 0x0 | <p>dfi_t_parin_lat</p> <p>Specifies the number of DFI PHY clock cycles between when the dfi_cs signal is asserted and when the associated dfi_parity_in signal is driven.</p> <p>Programming Mode: Quasi-dynamic Group 4</p> |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | <p>dfi_t_wrdata_delay</p> <p>Specifies the number of DFI clock cycles between when the dfi_wrdata_en signal is asserted and when the corresponding write data transfer is completed on the DRAM bus. This corresponds to the DFI timing parameter twrdata_delay. Refer to PHY specification for correct value.</p> <p>Value to be programmed is in terms of DFI clocks, not PHY clocks. In FREQ_RATIO=2, divide PHY's value by 2 and round up to next integer.</p> <p>Programming Mode: Quasi-dynamic Group 4</p> |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x04 | <p>dfi_t_dram_clk_disable</p> <p>Specifies the number of DFI clock cycles from the assertion of the dfi_dram_clk_disable signal on the DFI until the clock to the DRAM memory devices, at the PHY-DRAM boundary, maintains a low value. If the DFI clock and the memory clock are not phase aligned, this timing parameter should be rounded up to the next integer value.</p> <p>Programming Mode: Quasi-dynamic Group 4</p> |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | <p>dfi_t_dram_clk_enable</p> <p>Specifies the number of DFI clock cycles from the deassertion of the dfi_dram_clk_disable signal on the DFI until the first valid rising edge of the clock to the DRAM memory devices, at the PHY-DRAM boundary. If the DFI clock and the memory clock are not phase aligned, this timing parameter should be rounded up to the next integer value.</p> <p>Programming Mode: Quasi-dynamic Group 4</p> |

DDRC DFILPCFG0

Address: Operational Base + offset (0x0198)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|-------------|
| 31:29 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 28:24 | RW | 0x07 | <p>dfi_tlp_resp Setting in DFI clock cycles for DFI's tlp_resp time. Same value is used for both Power Down, Self Refresh, Deep Power Down and Maximum Power Saving modes. DFI 2.1 specification onwards, recommends using a fixed value of 7 always. Programming Mode: Static</p> |
| 23:20 | RW | 0x0 | <p>dfi_lp_wakeup_dpd Value in DFI clock cycles to drive on dfi_lp_wakeup signal when Deep Power Down mode is entered. Determines the DFI's tlp_wakeup time: 0x0 - 16 cycles 0x1 - 32 cycles 0x2 - 64 cycles 0x3 - 128 cycles 0x4 - 256 cycles 0x5 - 512 cycles 0x6 - 1024 cycles 0x7 - 2048 cycles 0x8 - 4096 cycles 0x9 - 8192 cycles 0xA - 16384 cycles 0xB - 32768 cycles 0xC - 65536 cycles 0xD - 131072 cycles 0xE - 262144 cycles 0xF - Unlimited This is only present for designs supporting mDDR or LPDDR2/LPDDR3 devices. Programming Mode: Static</p> |
| 19:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | <p>dfi_lp_en_dpd Enables DFI Low Power interface handshaking during Deep Power Down Entry/Exit. 0 - Disabled; 1 - Enabled This is only present for designs supporting mDDR or LPDDR2/LPDDR3 devices. Programming Mode: Static</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 15:12 | RW | 0x0 | <p>dfi_lp_wakeup_sr Value in DFI clpck cycles to drive on dfi_lp_wakeup signal when Self Refresh mode is entered. Determines the DFI's tlp_wakeup time: 0x0 - 16 cycles 0x1 - 32 cycles 0x2 - 64 cycles 0x3 - 128 cycles 0x4 - 256 cycles 0x5 - 512 cycles 0x6 - 1024 cycles 0x7 - 2048 cycles 0x8 - 4096 cycles 0x9 - 8192 cycles 0xA - 16384 cycles 0xB - 32768 cycles 0xC - 65536 cycles 0xD - 131072 cycles 0xE - 262144 cycles 0xF - Unlimited Programming Mode: Static</p> |
| 11:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | <p>dfi_lp_en_sr Enables DFI Low Power interface handshaking during Self Refresh Entry/Exit. 0 - Disabled; 1 - Enabled Programming Mode: Static</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7:4 | RW | 0x0 | <p>dfi_lp_wakeup_pd Value in DFI clock cycles to drive on dfi_lp_wakeup signal when Power Down mode is entered. Determines the DFI's tlp_wakeup time:</p> <ul style="list-style-type: none"> 0x0 - 16 cycles 0x1 - 32 cycles 0x2 - 64 cycles 0x3 - 128 cycles 0x4 - 256 cycles 0x5 - 512 cycles 0x6 - 1024 cycles 0x7 - 2048 cycles 0x8 - 4096 cycles 0x9 - 8192 cycles 0xA - 16384 cycles 0xB - 32768 cycles 0xC - 65536 cycles 0xD - 131072 cycles 0xE - 262144 cycles 0xF - Unlimited <p>Programming Mode: Static</p> |
| 3:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | <p>dfi_lp_en_pd Enables DFI Low Power interface handshaking during Power Down Entry/Exit. 0: Disabled; 1: Enabled Programming Mode: Static</p> |

DDRC DFILPCFG1

Address: Operational Base + offset (0x019c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 7:4 | RW | 0x0 | <p>dfi_lp_wakeup_mpsm Value in DFI clock cycles to drive on dfi_lp_wakeup signal when Maximum Power Saving Mode is entered. Determines the DFI's tlp_wakeup time: 0x0 - 16 cycles 0x1 - 32 cycles 0x2 - 64 cycles 0x3 - 128 cycles 0x4 - 256 cycles 0x5 - 512 cycles 0x6 - 1024 cycles 0x7 - 2048 cycles 0x8 - 4096 cycles 0x9 - 8192 cycles 0xA - 16384 cycles 0xB - 32768 cycles 0xC - 65536 cycles 0xD - 131072 cycles 0xE - 262144 cycles 0xF - Unlimited This is only present for designs supporting DDR devices. Programming Mode: Static</p> |
| 3:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | <p>dfi_lp_en_mpsm Enables DFI Low Power interface handshaking during Maximum Power Saving Mode Entry/Exit. 0 - Disabled; 1 - Enabled This is only present for designs supporting DDR devices. Programming Mode: Static</p> |

DDRC_DFIUPD0

Address: Operational Base + offset (0x01a0)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 31 | RW | 0x0 | <p>dis_auto_ctrlupd When '1', disable the automatic dfi_ctrlupd_req generation by the uMCTL2. The core must issue the dfi_ctrlupd_req signal using register DBGCMD.ctrlupd. When '0', uMCTL2 issues dfi_ctrlupd_req periodically. Programming Mode: Quasi-dynamic Group 3</p> |
| 30 | RW | 0x0 | <p>dis_auto_ctrlupd_srx When '1', disable the automatic dfi_ctrlupd_req generation by the uMCTL2 at self-refresh exit. When '0', uMCTL2 issues a dfi_ctrlupd_req before or after exiting self-refresh, depending on DFIUPD0.ctrlupd_pre_srx. Programming Mode: Static</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | <p>ctrlupd_pre_srx Selects dfi_ctrlupd_req requirements at SRX: 0 : send ctrlupd after SRX 1 : send ctrlupd before SRX If DFIUPD0.dis_auto_ctrlupd_srx=1, this register has no impact, because no dfi_ctrlupd_req will be issued when SRX. Programming Mode: Static</p> |
| 28:26 | RO | 0x0 | reserved |
| 25:16 | RW | 0x040 | <p>dfi_t_ctrlup_max Specifies the maximum number of DFI clock cycles that the dfi_ctrlupd_req signal can assert. Lowest value to assign to this variable is 0x40. Programming Mode: Static</p> |
| 15:10 | RO | 0x0 | reserved |
| 9:0 | RW | 0x003 | <p>dfi_t_ctrlup_min Specifies the minimum number of DFI clock cycles that the dfi_ctrlupd_req signal must be asserted. The uMCTL2 expects the PHY to respond within this time. If the PHY does not respond, the uMCTL2 will de-assert dfi_ctrlupd_req after dfi_t_ctrlup_min + 2 cycles. Lowest value to assign to this variable is 0x3. Programming Mode: Static</p> |

DDRC DFIUPD1

Address: Operational Base + offset (0x01a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x00 | <p>dfi_t_ctrlupd_interval_min_x1024 This is the minimum amount of time between uMCTL2 initiated DFI update requests (which is executed whenever the uMCTL2 is idle). Set this number higher to reduce the frequency of update requests, which can have a small impact on the latency of the first read request when the uMCTL2 is idle. Minimum allowed value for this field is 1. Unit: 1024 DFI clock cycles Programming Mode: Static</p> |
| 15:8 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7:0 | RW | 0x01 | <p>dfi_t_ctrlupd_interval_max_x1024 This is the maximum amount of time between uMCTL2 initiated DFI update requests. This timer resets with each update request; when the timer expires dfi_ctrlupd_req is sent and traffic is blocked until the dfi_ctrlupd_ackx is received. PHY can use this idle time to recalibrate the delay lines to the DLLs. The DFI controller update is also used to reset PHY FIFO pointers in case of data capture errors. Updates are required to maintain calibration over PVT, but frequent updates may impact performance. Minimum allowed value for this field is 1.</p> <p>Note: Value programmed for DFIUPD1.dfi_t_ctrlupd_interval_max_x1024 must be greater than DFIUPD1.dfi_t_ctrlupd_interval_min_x1024.</p> <p>Unit: 1024 DFI clock cycles Programming Mode: Static</p> |

DDRC_DFIUPD2

Address: Operational Base + offset (0x01a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | <p>dfi_phyupd_en Enables the support for acknowledging PHY-initiated updates: 0 - Disabled; 1 - Enabled Programming Mode: Static</p> |

DDRC_DFIMISC

Address: Operational Base + offset (0x01b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x00 | <p>dfi_frequency Indicates the operating frequency of the system. The number of supported frequencies and the mapping of signal values to clock frequencies are defined by the PHY. Programming Mode: Quasi-dynamic Group 1</p> |
| 7:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | <p>dfi_init_start PHY init start request signal. When asserted it triggers the PHY init start request Programming Mode: Quasi-dynamic Group 3</p> |
| 4 | RW | 0x0 | <p>ctl_idle_en Enables support of ctl_idle signal, which is non-DFI related pin specific to certain Synopsys PHYs. See signal description of ctl_idle signal for further details of ctl_idle functionality. Programming Mode: Static</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 3 | RW | 0x0 | <p>share_dfi_dram_clk_disable Indicate dfi_dram_clk_disable is shared for two channels or not. 1: Share mode 0: Not share.</p> <p>In Shared mode, Controller does not request PHY stop clock while any rank of any channel has not disable clock. Note: when dfi_dram_clk_disable is shared by two channels, an additional DFF is inserted, that will cause dram clock enable is delay one cycle.</p> <p>Suggest set dfi_t_dram_clk_enable value to Tdram_clk_enable+1. Tdram_clk_enable value is from PHY, which indicate how many cycles from dfi_dram_clk_disable de-assert to dram clock output.</p> <p>Programming Mode: Static</p> |
| 2 | RW | 0x0 | <p>dfi_data_cs_polarity Defines polarity of dfi_wrdata_cs and dfi_rddata_cs signals. 0: Signals are active low; 1: Signals are active high</p> <p>Programming Mode: Static</p> |
| 1 | RW | 0x0 | <p>phy_dbm_mode DBI implemented in DDRC or PHY. 0 - DDRC implements DBI functionality. 1 - PHY implements DBI functionality. Present only in designs configured to support DDR and LPDDR.</p> <p>Programming Mode: Static</p> |
| 0 | RW | 0x1 | <p>dfi_init_complete_en PHY initialization complete enable signal. When asserted the dfi_init_complete signal can be used to trigger SDRAM initialisation</p> <p>Programming Mode: Quasi-dynamic Group 3</p> |

DDRC DFITMG2

Address: Operational Base + offset (0x01b4)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:15 | RO | 0x0 | reserved |
| 14:8 | RW | 0x02 | <p>dfi_tphy_rdcslat Number of DFI PHY clock cycles between when a read command is sent on the DFI control interface and when the associated dfi_rddata_cs signal is asserted. This corresponds to the DFI timing parameter tphy_rdcslat. Refer to PHY specification for correct value.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |
| 7:6 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 5:0 | RW | 0x02 | <p>dfi_tphy_wrcslat Number of DFI PHY clock cycles between when a write command is sent on the DFI control interface and when the associated dfi_wrdata_cs signal is asserted. This corresponds to the DFI timing parameter tphy_wrcslat. Refer to PHY specification for correct value.</p> <p>Programming Mode: Quasi-dynamic Group 2 and Group 4</p> |

DDRC DFITMG3

Address: Operational Base + offset (0x01b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | <p>dfi_t_geardown_delay The delay from dfi_geardown_en assertion to the time of the PHY being ready to receive commands. Refer to PHY specification for correct value.</p> <p>When the controller is operating in 1:2 frequency ratio mode, program this to (tgeardown_delay/2) and round it up to the next integer value.</p> <p>Unit: DFI Clocks</p> <p>Programming Mode: Static</p> |

DDRC DFISTAT

Address: Operational Base + offset (0x01bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | <p>dfi_lp_ack Stores the value of the dfi_lp_ack input to the controller.</p> <p>Programming Mode: Dynamic</p> |
| 0 | RO | 0x0 | <p>dfi_init_complete The status flag register which announces when the DFI initialization has been completed. The DFI INIT triggered by dfi_init_start signal and then the dfi_init_complete flag is polled to know when the initialization is done.</p> <p>Programming Mode: Dynamic</p> |

DDRC DBICTL

Address: Operational Base + offset (0x01c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:3 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 2 | RW | 0x0 | <p>rd_dbi_en Read DBI enable signal in DDRC. 0 - Read DBI is disabled. 1 - Read DBI is enabled.</p> <p>This signal must be set the same value as DRAM's mode register. DDR: MR5 bit A12. When x4 devices are used, this signal must be set to 0. LPDDR: MR3[6]</p> <p>Programming Mode: Quasi-dynamic Group 1</p> |
| 1 | RW | 0x0 | <p>wr_dbi_en Write DBI enable signal in DDRC. 0 - Write DBI is disabled. 1 - Write DBI is enabled.</p> <p>This signal must be set the same value as DRAM's mode register. DDR: MR5 bit A11. When x4 devices are used, this signal must be set to 0. LPDDR: MR3[7]</p> <p>Programming Mode: Quasi-dynamic Group 1</p> |
| 0 | RW | 0x1 | <p>dm_en DM enable signal in DDRC. 0 - DM is disabled; 1 - DM is enabled.</p> <p>This signal must be set the same logical value as DRAM's mode register.</p> <p>DDR: Set this to same value as MR5 bit A10. When x4 devices are used, this signal must be set to 0. LPDDR: Set this to inverted value of MR13[5] which is opposite polarity from this signal</p> <p>Programming Mode: Quasi-dynamic Group 3</p> |

DDRC DFIPHYMSTR

Address: Operational Base + offset (0x01c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | <p>dfi_phymstr_en Enables the PHY Master Interface: 0 - Disabled; 1 - Enabled</p> <p>Programming Mode: Dynamic</p> |

DDRC ADDRMAP0

Address: Operational Base + offset (0x0200)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:21 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 20:16 | RW | 0x00 | <p>addrmap_dch_bit0 Selects the HIF address bit used as data channel address bit 0. Valid Range: 0 to 30, and 31 (Traffic constraints apply based on the register value when UMCTL2_EXCL_ACCESS>0. See Exclusive Access section for details.) Internal Base: 2 The selected address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then channel bit is set to 0. Programming Mode: Static</p> |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x00 | <p>addrmap_cs_bit1 Selects the HIF address bit used as rank address bit 1. Valid Range: 0 to 28, and 31 Internal Base: 7 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then rank address bit 1 is set to 0. Programming Mode: Static</p> |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | <p>addrmap_cs_bit0 Selects the HIF address bit used as rank address bit 0. Valid Range: 0 to 29, and 31 Internal Base: 6 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then rank address bit 0 is set to 0. Programming Mode: Static</p> |

DDRC_ADDRMAP1

Address: Operational Base + offset (0x0204)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x00 | <p>addrmap_bank_b2 Selects the HIF address bit used as bank address bit 2. Valid Range: 0 to 31 and 63 Internal Base: 4 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank address bit 2 is set to 0. Programming Mode: Static</p> |
| 15:14 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 13:8 | RW | 0x00 | <p>addrmap_bank_b1 Selects the HIF address bits used as bank address bit 1. Valid Range: 0 to 32 and 63 Internal Base: 3 The selected HIF address bit for each of the bank address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank address bit 1 is set to 0. Programming Mode: Static</p> |
| 7:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x00 | <p>addrmap_bank_b0 Selects the HIF address bits used as bank address bit 0. Valid Range: 0 to 32 and 63 Internal Base: 2 The selected HIF address bit for each of the bank address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank address bit 0 is set to 0. Programming Mode: Static</p> |

DDRC_ADDRMAP2

Address: Operational Base + offset (0x0208)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:24 | RW | 0x0 | <p>addrmap_col_b5 Full bus width mode: Selects the HIF address bit used as column address bit 5. Half bus width mode: Selects the HIF address bit used as column address bit 6. Valid Range: 0 to 7, and 15 Internal Base: 5 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. Programming Mode: Static</p> |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RW | 0x0 | <p>addrmap_col_b4 Full bus width mode: Selects the HIF address bit used as column address bit 4. Half bus width mode: Selects the HIF address bit used as column address bit 5. Valid Range: 0 to 7, and 15 Internal Base: 4 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0. Programming Mode: Static</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 15:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x0 | <p>addrmap_col_b3 Full bus width mode: Selects the HIF address bit used as column address bit 3. Half bus width mode: Selects the HIF address bit used as column address bit 4. Valid Range: 0 to 7 Internal Base: 3 The selected HIF address bit is determined by adding the internal base to the value of this field. Note, if UMCTL2_INCL_ARB=1, MEMC_BURST_LENGTH=16, Full bus width (MSTR.data_bus_width==00) and BL16 (MSTR.burst_rdwr=1000), it is recommended to program this to 0. Programming Mode: Static</p> |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>addrmap_col_b2 Full bus width mode: Selects the HIF address bit used as column address bit 2. Half bus width mode: Selects the HIF address bit used as column address bit 3. Valid Range: 0 to 7 Internal Base: 2 The selected HIF address bit is determined by adding the internal base to the value of this field. Note, if UMCTL2_INCL_ARB=1 and MEMC_BURST_LENGTH=8, it is required to program this to 0 unless: in Half or Quarter bus width (MSTR.data_bus_width!=00) and PCCFG.bl_exp_mode==1 and either In DDR and ADDRMAP8.addrmap_bg_b0==0 or In LPDDR and ADDRMAP1.addrmap_bank_b0==0 If UMCTL2_INCL_ARB=1 and MEMC_BURST_LENGTH=16, it is required to program this to 0 unless: in Half or Quarter bus width (MSTR.data_bus_width!=00) and PCCFG.bl_exp_mode==1 and In DDR and ADDRMAP8.addrmap_bg_b0==0 Otherwise, if MEMC_BURST_LENGTH=8 and Full Bus Width (MSTR.data_bus_width==00), it is recommended to program this to 0 so that HIF[2] maps to column address bit 2. If MEMC_BURST_LENGTH=16 and Full Bus Width (MSTR.data_bus_width==00), it is recommended to program this to 0 so that HIF[2] maps to column address bit 2. If MEMC_BURST_LENGTH=16 and Half Bus Width (MSTR.data_bus_width==01), it is recommended to program this to 0 so that HIF[2] maps to column address bit 3. Programming Mode: Static</p> |

DDRC_ADDRMAP3

Address: Operational Base + offset (0x020c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28:24 | RW | 0x00 | <p>addrmap_col_b9 Full bus width mode: Selects the HIF address bit used as column address bit 9. Half bus width mode: Selects the HIF address bit used as column address bit 11 (10 in LPDDR2/LPDDR3 mode). Valid Range: 0 to 7, and 31. Internal Base: 9</p> <p>The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then this column address bit is set to 0.</p> <p>Note: Per JEDEC DDR2/3/mDDR specification, column address bit 10 is reserved for indicating auto-precharge, and hence no source address bit can be mapped to column address bit 10. In LPDDR2/LPDDR3, there is a dedicated bit for autoprecharge in the CA bus and hence column bit 10 is used.</p> <p>Programming Mode: Static</p> |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | <p>addrmap_col_b8 Full bus width mode: Selects the HIF address bit used as column address bit 8. Half bus width mode: Selects the HIF address bit used as column address bit 9. Valid Range: 0 to 7, and 31. Internal Base: 8</p> <p>The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then this column address bit is set to 0.</p> <p>Note: Per JEDEC DDR2/3/mDDR specification, column address bit 10 is reserved for indicating auto-precharge, and hence no source address bit can be mapped to column address bit 10. In LPDDR2/LPDDR3, there is a dedicated bit for autoprecharge in the CA bus and hence column bit 10 is used.</p> <p>Programming Mode: Static</p> |
| 15:13 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12:8 | RW | 0x00 | <p>addrmap_col_b7</p> <p>Full bus width mode: Selects the HIF address bit used as column address bit 7.</p> <p>Half bus width mode: Selects the HIF address bit used as column address bit 8.</p> <p>Valid Range: 0 to 7, and 31.</p> <p>Internal Base: 7</p> <p>The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then this column address bit is set to 0.</p> <p>Programming Mode: Static</p> |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>addrmap_col_b6</p> <p>Full bus width mode: Selects the HIF address bit used as column address bit 6.</p> <p>Half bus width mode: Selects the HIF address bit used as column address bit 7.</p> <p>Valid Range: 0 to 7, and 15</p> <p>Internal Base: 6</p> <p>The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then this column address bit is set to 0.</p> <p>Programming Mode: Static</p> |

DDRC ADDRMAP4

Address: Operational Base + offset (0x0210)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x00 | <p>addrmap_col_b11</p> <p>Full bus width mode: Selects the HIF address bit used as column address bit 13 (11 in LPDDR2/LPDDR3 mode).</p> <p>Half bus width mode: Unused. To make it unused, this should be tied to 4'hF.</p> <p>Valid Range: 0 to 7, and 31.</p> <p>Internal Base: 11</p> <p>The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then this column address bit is set to 0.</p> <p>Note: Per JEDEC DDR2/3/mDDR specification, column address bit 10 is reserved for indicating auto-precharge, and hence no source address bit can be mapped to column address bit 10. In LPDDR2/LPDDR3, there is a dedicated bit for autoprecharge in the CA bus and hence column bit 10 is used.</p> <p>Programming Mode: Static</p> |
| 7:5 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 4:0 | RW | 0x00 | <p>addrmap_col_b10</p> <p>Full bus width mode: Selects the HIF address bit used as column address bit 11 (10 in LPDDR2/LPDDR3 mode).</p> <p>Half bus width mode: Selects the HIF address bit used as column address bit 13 (11 in LPDDR2/LPDDR3 mode).</p> <p>Valid Range: 0 to 7, and 31.</p> <p>Internal Base: 10</p> <p>The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 31 and then this column address bit is set to 0.</p> <p>Note: Per JEDEC DDR2/3/mDDR specification, column address bit 10 is reserved for indicating auto-precharge, and hence no source address bit can be mapped to column address bit 10. In LPDDR2/LPDDR3, there is a dedicated bit for autoprecharge in the CA bus and hence column bit 10 is used.</p> <p>Programming Mode: Static</p> |

DDRC ADDRMAPS

Address: Operational Base + offset (0x0214)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:24 | RW | 0x0 | <p>addrmap_row_b11</p> <p>Selects the HIF address bit used as row address bit 11.</p> <p>Valid Range: 0 to 11, and 15</p> <p>Internal Base: 17</p> <p>The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then row address bit 11 is set to 0.</p> <p>Programming Mode: Static</p> |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RW | 0x0 | <p>addrmap_row_b2_10</p> <p>Selects the HIF address bits used as row address bits 2 to 10.</p> <p>Valid Range: 0 to 11, and 15</p> <p>Internal Base: 8 (for row address bit 2), 9 (for row address bit 3), 10 (for row address bit 4) etc increasing to 16 (for row address bit 10)</p> <p>The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. When value 15 is used the values of row address bits 2 to 10 are defined by registers ADDRMAP9, ADDRMAP10, ADDRMAP11.</p> <p>Programming Mode: Static</p> |
| 15:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:8 | RW | 0x0 | <p>addrmap_row_b1 Selects the HIF address bits used as row address bit 1. Valid Range: 0 to 11 Internal Base: 7 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. Programming Mode: Static</p> |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>addrmap_row_b0 Selects the HIF address bits used as row address bit 0. Valid Range: 0 to 11 Internal Base: 6 The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. Programming Mode: Static</p> |

DDRC ADDRMAP6

Address: Operational Base + offset (0x0218)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | <p>lpddr3_6gb_12gb Set this to 1 if there is an LPDDR3 SDRAM 6Gb or 12Gb device in use. 1 - LPDDR3 SDRAM 6Gb/12Gb device in use. Every address having row[14:13]==2'b11 is considered as invalid 0 - non-LPDDR3 6Gb/12Gb device in use. All addresses are valid Present only in designs configured to support LPDDR3. Programming Mode: Static</p> |
| 30:29 | RW | 0x0 | <p>lpDDR_6gb_12gb_24gb Indicates what type of LPDDR SDRAM device is in use. 2'b00: No LPDDR SDRAM 6Gb/12Gb/24Gb device in use. All addresses are valid 2'b01: LPDDR SDRAM 6Gb device in use. Every address having row[14:13]==2'b11 is considered as invalid 2'b10: LPDDR SDRAM 12Gb device in use. Every address having row[15:14]==2'b11 is considered as invalid 2'b11: LPDDR SDRAM 24Gb device in use. Unsupported Present only in designs configured to support LPDDR. Programming Mode: Static</p> |
| 28 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 27:24 | RW | 0x0 | <p>addrmap_row_b15 Selects the HIF address bit used as row address bit 15. Valid Range: 0 to 11, and 15 Internal Base: 21 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then row address bit 15 is set to 0. Programming Mode: Static</p> |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RW | 0x0 | <p>addrmap_row_b14 Selects the HIF address bit used as row address bit 14. Valid Range: 0 to 11, and 15 Internal Base: 20 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then row address bit 14 is set to 0. Programming Mode: Static</p> |
| 15:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x0 | <p>addrmap_row_b13 Selects the HIF address bit used as row address bit 13. Valid Range: 0 to 11, and 15 Internal Base: 19 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then row address bit 13 is set to 0. Programming Mode: Static</p> |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>addrmap_row_b12 Selects the HIF address bit used as row address bit 12. Valid Range: 0 to 11, and 15 Internal Base: 18 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then row address bit 12 is set to 0. Programming Mode: Static</p> |

DDRC ADDRMAP7

Address: Operational Base + offset (0x021c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11:8 | RW | 0x0 | <p>addrmap_row_b17 Selects the HIF address bit used as row address bit 17. Valid Range: 0 to 11, and 15 Internal Base: 23 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then row address bit 17 is set to 0. Programming Mode: Static</p> |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>addrmap_row_b16 Selects the HIF address bit used as row address bit 16. Valid Range: 0 to 11, and 15 Internal Base: 22 The selected HIF address bit is determined by adding the internal base to the value of this field. If unused, set to 15 and then row address bit 16 is set to 0. Programming Mode: Static</p> |

DDRC_ADDRMAP8

Address: Operational Base + offset (0x0220)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | <p>addrmap_bg_b1 Selects the HIF address bits used as bank group address bit 1. Valid Range: 0 to 32, and 63 Internal Base: 3 The selected HIF address bit for each of the bank group address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank group address bit 1 is set to 0. Programming Mode: Static</p> |
| 7:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x00 | <p>addrmap_bg_b0 Selects the HIF address bits used as bank group address bit 0. Valid Range: 0 to 32 and 63 Internal Base: 2 The selected HIF address bit for each of the bank group address bits is determined by adding the internal base to the value of this field. If unused, set to 63 and then bank group address bit 0 is set to 0. Programming Mode: Static</p> |

DDRC_ADDRMAP9

Address: Operational Base + offset (0x0224)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| | | | addrmap_row_b5 Selects the HIF address bits used as row address bit 5. Valid Range: 0 to 11 Internal Base: 11 |
| 27:24 | RW | 0x0 | The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15. Programming Mode: Static |
| 23:20 | RO | 0x0 | reserved |
| | | | addrmap_row_b4 Selects the HIF address bits used as row address bit 4. Valid Range: 0 to 11 Internal Base: 10 |
| 19:16 | RW | 0x0 | The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15. Programming Mode: Static |
| 15:12 | RO | 0x0 | reserved |
| | | | addrmap_row_b3 Selects the HIF address bits used as row address bit 3. Valid Range: 0 to 11 Internal Base: 9 |
| 11:8 | RW | 0x0 | The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15. Programming Mode: Static |
| 7:4 | RO | 0x0 | reserved |
| | | | addrmap_row_b2 Selects the HIF address bits used as row address bit 2. Valid Range: 0 to 11 Internal Base: 8 |
| 3:0 | RW | 0x0 | The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15. Programming Mode: Static |

DDRC ADDRMAP10

Address: Operational Base + offset (0x0228)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:28 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 27:24 | RW | 0x0 | <p>addrmap_row_b9 Selects the HIF address bits used as row address bit 9. Valid Range: 0 to 11 Internal Base: 15</p> <p>The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field.</p> <p>This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15.</p> <p>Programming Mode: Static</p> |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RW | 0x0 | <p>addrmap_row_b8 Selects the HIF address bits used as row address bit 8. Valid Range: 0 to 11 Internal Base: 14</p> <p>The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field.</p> <p>This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15.</p> <p>Programming Mode: Static</p> |
| 15:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x0 | <p>addrmap_row_b7 Selects the HIF address bits used as row address bit 7. Valid Range: 0 to 11 Internal Base: 13</p> <p>The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field.</p> <p>This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15.</p> <p>Programming Mode: Static</p> |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>addrmap_row_b6 Selects the HIF address bits used as row address bit 6. Valid Range: 0 to 11 Internal Base: 12</p> <p>The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field.</p> <p>This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15.</p> <p>Programming Mode: Static</p> |

DDRC ADDRMAP11

Address: Operational Base + offset (0x022c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:4 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 3:0 | RW | 0x0 | <p>addrmap_row_b10 Selects the HIF address bits used as row address bit 10. Valid Range: 0 to 11 Internal Base: 16</p> <p>The selected HIF address bit for each of the row address bits is determined by adding the internal base to the value of this field. This register field is used only when ADDRMAP5.addrmap_row_b2_10 is set to value 15. Programming Mode: Static</p> |

DDRC_ODTCFG

Address: Operational Base + offset (0x0240)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:24 | RW | 0x4 | <p>wr_odt_hold DFI PHY clock cycles to hold ODT for a write command. The minimum supported value is 2. Recommended values: DDR2: BL8: 0x5 (DDR2-400/533/667), 0x6 (DDR2-800), 0x7(DDR2-1066); BL4: 0x3 (DDR2-400/533/667), 0x4 (DDR2-800), 0x5 (DDR2-1066) DDR3: BL8: 0x6 DDR: BL8: 5 + WR_PREAMBLE + CRC_MODE. WR_PREAMBLE = 1 (1tCK write preamble), 2 (2tCK write preamble). CRC_MODE = 0 (not CRC mode), 1 (CRC mode) LPDDR3: BL8: 7 + RU(tODTon(max)/tCK) Programming Mode: Quasi-dynamic Group 1 and Group 4</p> |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | <p>wr_odt_delay The delay, in DFI PHY clock cycles, from issuing a write command to setting ODT values associated with that command. ODT setting must remain constant for the entire time that DQS is driven by the uMCTL2. Recommended values: DDR2: CWL + AL - 3 (DDR2-400/533/667), CWL + AL - 4 (DDR2-800), CWL + AL - 5 (DDR2-1066). If (CWL + AL - 3 < 0), uMCTL2 does not support ODT for write operation. DDR3: 0x0 DDR: DFITMG1.dfi_t_cmd_lat (to adjust for CAL mode) LPDDR3: WL - 1 - RU(tODTon(max)/tCK)) Programming Mode: Quasi-dynamic Group 1 and Group 4</p> |
| 15:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 11:8 | RW | 0x4 | <p>rd_odt_hold DFI PHY clock cycles to hold ODT for a read command. The minimum supported value is 2. Recommended values: DDR2: BL8: 0x6 (not DDR2-1066), 0x7 (DDR2-1066); BL4: 0x4 (not DDR2-1066), 0x5 (DDR2-1066) DDR3: BL8 - 0x6 DDR: BL8: 5 + RD_PREAMBLE. RD_PREAMBLE = 1 (1tCK write preamble), 2 (2tCK write preamble) LPDDR3: BL8: 5 + RU(tDQSCK(max)/tCK) - RD(tDQSCK(min)/tCK) + RU(tODTon(max)/tCK) Programming Mode: Quasi-dynamic Group 1 and Group 4</p> |
| 7 | RO | 0x0 | reserved |
| 6:2 | RW | 0x00 | <p>rd_odt_delay The delay, in DFI PHY clock cycles, from issuing a read command to setting ODT values associated with that command. ODT setting must remain constant for the entire time that DQS is driven by the uMCTL2. Recommended values: DDR2: CL + AL - 4 (not DDR2-1066), CL + AL - 5 (DDR2-1066). If (CL + AL - 4 < 0), uMCTL2 does not support ODT for read operation. DDR3: CL - CWL DDR: CL - CWL - RD_PREAMBLE + WR_PREAMBLE + DFITMG1.dfi_t_cmd_lat (to adjust for CAL mode). WR_PREAMBLE = 1 (1tCK write preamble), 2 (2tCK write preamble). RD_PREAMBLE = 1 (1tCK write preamble), 2 (2tCK write preamble). If (CL - CWL - RD_PREAMBLE + WR_PREAMBLE) < 0, uMCTL2 does not support ODT for read operation. LPDDR3: RL + RD(tDQSCK(min)/tCK) - 1 - RU(tODTon(max)/tCK) Programming Mode: Quasi-dynamic Group 1 and Group 4</p> |
| 1:0 | RO | 0x0 | reserved |

DDRC ODTMAP

Address: Operational Base + offset (0x0244)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:14 | RO | 0x0 | reserved |
| 13:12 | RW | 0x2 | <p>rank1_rd_odt Indicates which remote ODTs must be turned on during a read from rank 1. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Programming Mode: Static</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:10 | RO | 0x0 | reserved |
| 9:8 | RW | 0x2 | rank1_wr_odt Indicates which remote ODTs must be turned on during a write to rank 1. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Programming Mode: Static |
| 7:6 | RO | 0x0 | reserved |
| 5:4 | RW | 0x1 | rank0_rd_odt Indicates which remote ODTs must be turned on during a read from rank 0. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Programming Mode: Static |
| 3:2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x1 | rank0_wr_odt Indicates which remote ODTs must be turned on during a write to rank 0. Each rank has a remote ODT (in the SDRAM) which can be turned on by setting the appropriate bit here. Rank 0 is controlled by the LSB; rank 1 is controlled by bit next to the LSB, etc. For each rank, set its bit to 1 to enable its ODT. Programming Mode: Static |

DDRC_SCHED

Address: Operational Base + offset (0x0250)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30:24 | RW | 0x00 | rdwr_idle_gap When the preferred transaction store is empty for these many clock cycles, switch to the alternate transaction store if it is non-empty. The read transaction store (both high and low priority) is the default preferred transaction store and the write transaction store is the alternative store. When prefer write over read is set this is reversed. 0x0 is a legal value for this register. When set to 0x0, the transaction store switching will happen immediately when the switching conditions become true. Programming Mode: Static |
| 23:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:8 | RW | 0x8 | <p>lpr_num_entries</p> <p>Number of entries in the low priority transaction store is this value + 1. (MEMC_NO_OF_ENTRY - (SCED1.lpr_num_entries + 1)) is the number of entries available for the high priority transaction store. Setting this to maximum value allocates all entries to low priority transaction store. Setting this to 0 allocates 1 entry to low priority transaction store and the rest to high priority transaction store.</p> <p>Programming Mode: Static</p> |
| 7:3 | RO | 0x0 | reserved |
| 2 | RW | 0x1 | <p>pageclose</p> <p>If true, bank is kept open only while there are page hit transactions available in the CAM to that bank. The last read or write command in the CAM with a bank and page hit will be executed with auto-precharge if SCED1.pageclose_timer=0. Even if this register set to 1 and SCED1.pageclose_timer is set to 0, explicit precharge (and not auto-precharge) may be issued in some cases where there is a mode switch between Write and Read or between LPR and HPR. The Read and Write commands that are executed as part of the ECC scrub requests are also executed without auto-precharge.</p> <p>If false, the bank remains open until there is a need to close it (to open a different page, or for page timeout or refresh timeout) - also known as open page policy.</p> <p>The pageclose feature provides a midway between Open and Close page policies.</p> <p>Programming Mode: Quasi-dynamic Group 3</p> |
| 1:0 | RO | 0x0 | reserved |

DDRC_SCHED1

Address: Operational Base + offset (0x0254)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 7:0 | RW | 0x00 | <p>pageclose_timer</p> <p>This field works in conjunction with SCHED.pageclose. It only has meaning if SCHED.pageclose==1. If SCHED.pageclose==1 and pageclose_timer==0, then an auto-precharge may be scheduled for last read or write command in the CAM with a bank and page hit. Note, sometimes an explicit precharge is scheduled instead of the auto-precharge. See SCHED.pageclose for details of when this may happen. If SCHED.pageclose==1 and pageclose_timer>0, then an auto-precharge is not scheduled for last read or write command in the CAM with a bank and page hit. Instead, a timer is started, with pageclose_timer as the initial value. There is a timer on a per bank basis. The timer decrements unless the next read or write in the CAM to a bank is a page hit. It gets reset to pageclose_timer value if the next read or write in the CAM to a bank is a page hit. Once the timer has reached zero, an explicit precharge will be attempted to be scheduled.</p> <p>Programming Mode: Static</p> |

DDRC_PERFLPR1

Address: Operational Base + offset (0x0264)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:24 | RW | 0x0f | <p>lpr_xact_run_length</p> <p>Number of transactions that are serviced once the LPR queue goes critical is the smaller of:</p> <ul style="list-style-type: none"> (a) This number (b) Number of transactions available. <p>Unit: Transaction.</p> <p>Programming Mode: Quasi-dynamic Group 3</p> |
| 23:16 | RO | 0x0 | reserved |
| 15:0 | RW | 0x007f | <p>lpr_max_starve</p> <p>Number of DFI clocks that the LPR queue can be starved before it goes critical. The minimum valid functional value for this register is 0x1. Programming it to 0x0 will disable the starvation functionality; during normal operation, this function should not be disabled as it will cause excessive latencies.</p> <p>Programming Mode: Quasi-dynamic Group 3</p> |

DDRC_PERFWR1

Address: Operational Base + offset (0x026c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0x0f | w_xact_run_length Number of transactions that are serviced once the WR queue goes critical is the smaller of: (a) This number (b) Number of transactions available. Unit: Transaction. Programming Mode: Quasi-dynamic Group 3 |
| 23:16 | RO | 0x0 | reserved |
| 15:0 | RW | 0x007f | w_max_starve Number of DFI clocks that the WR queue can be starved before it goes critical. The minimum valid functional value for this register is 0x1. Programming it to 0x0 will disable the starvation functionality; during normal operation, this function should not be disabled as it will cause excessive latencies. Programming Mode: Quasi-dynamic Group 3 |

DDRC_DBG0

Address: Operational Base + offset (0x0300)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | dis_collision_page_opt When this is set to '0', auto-precharge is disabled for the flushed command in a collision case. Collision cases are write followed by read to same address, read followed by write to same address, or write followed by write to same address with DBG0.dis_wc bit = 1 (where same address comparisons exclude the two address bits representing critical word). Programming Mode: Static |
| 3:0 | RO | 0x0 | reserved |

DDRC_DBG1

Address: Operational Base + offset (0x0304)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | dis_hif When 1, uMCTL2 asserts the HIF command signal hif_cmd_stall. uMCTL2 will ignore the hif_cmd_valid and all other associated request signals. This bit is intended to be switched on-the-fly. Programming Mode: Dynamic |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | RW | 0x0 | <p>dis_dq</p> <p>When 1, uMCTL2 will not de-queue any transactions from the CAM. Bypass is also disabled. All transactions are queued in the CAM. No reads or writes are issued to SDRAM as long as this is asserted. This bit may be used to prevent reads or writes being issued by the uMCTL2, which makes it safe to modify certain register fields associated with reads and writes (see User Guide for details). After setting this bit, it is strongly recommended to poll DBGCAM.wr_data_pipeline_empty and DBGCAM.rd_data_pipeline_empty, before making changes to any registers which affect reads and writes. This will ensure that the relevant logic in the DDRC is idle. This bit is intended to be switched on-the-fly.</p> <p>Programming Mode: Dynamic</p> |

DDRC DBGCAM

Address: Operational Base + offset (0x0308)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29 | RO | 0x0 | <p>wr_data_pipeline_empty</p> <p>When 1, indicates that the write data pipeline on the DFI interface is empty. This register is intended to be polled at least twice after setting DBG1.dis_dq, to ensure that all remaining commands/data have completed.</p> <p>Programming Mode: Dynamic</p> |
| 28 | RO | 0x0 | <p>rd_data_pipeline_empty</p> <p>When 1, indicates that the read data pipeline on the DFI interface is empty. This register is intended to be polled at least twice after setting DBG1.dis_dq, to ensure that all remaining commands/data have completed.</p> <p>Programming Mode: Dynamic</p> |
| 27 | RO | 0x0 | reserved |
| 26 | RO | 0x0 | <p>dbg_wr_q_empty</p> <p>When 1, all the Write command queues and Write data buffers inside DDRC are empty. This register is to be used for debug purpose. An example use-case scenario: When Controller enters SelfRefresh using the Low-Power entry sequence, Controller is expected to have executed all the commands in its queues and the write and read data drained. Hence this register should be 1 at that time.</p> <p>Programming Mode: Dynamic</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 25 | RO | 0x0 | dbg_rd_q_empty When 1, all the Read command queues and Read data buffers inside DDRC are empty. This register is to be used for debug purpose. An example use-case scenario: When Controller enters SelfRefresh using the Low-Power entry sequence, Controller is expected to have executed all the commands in its queues and the write and read data drained. Hence this register should be 1 at that time. Programming Mode: Dynamic |
| 24 | RO | 0x0 | dbg_stall Stall Programming Mode: Dynamic |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RO | 0x0 | dbg_w_q_depth Write queue depth The last entry of WR queue is reserved for ECC SCRUB operation. This entry is not included in the calculation of the queue depth. Programming Mode: Dynamic |
| 15:12 | RO | 0x0 | reserved |
| 11:8 | RO | 0x0 | dbg_lpr_q_depth Low priority read queue depth. The last entry of Lpr queue is reserved for ECC SCRUB operation. This entry is not included in the calculation of the queue depth. Programming Mode: Dynamic |
| 7:0 | RO | 0x0 | reserved |

DDRC DBGCMD

Address: Operational Base + offset (0x030c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | ctrlupd Setting this register bit to 1 indicates to the uMCTL2 to issue a dfi_ctrlupd_req to the PHY. When this request is stored in the uMCTL2, the bit is automatically cleared. This operation must only be performed when DFIUPD0.dis_auto_ctrlupd=1. Programming Mode: Dynamic |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 4 | RW | 0x0 | <p>zq_calib_short</p> <p>Setting this register bit to 1 indicates to the uMCTL2 to issue a ZQCS (ZQ calibration short)/MPC(ZQ calibration) command to the SDRAM. When this request is stored in the uMCTL2, the bit is automatically cleared. This operation can be performed only when ZQCTL0.dis_auto_zq=1. It is recommended NOT to set this register bit if in Init operating mode. This register bit is ignored when in SelfRefresh(except LPDDR) and SR-Powerdown(LPDDR) and Deep power-down operating modes and Maximum Power Saving Mode.</p> <p>Programming Mode: Dynamic</p> |
| 3:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | <p>rank1_refresh</p> <p>Setting this register bit to 1 indicates to the uMCTL2 to issue a refresh to rank 1. Writing to this bit causes DBGSTAT.rank1_refresh_busy to be set. When DBGSTAT.rank1_refresh_busy is cleared, the command has been stored in uMCTL2.</p> <p>This operation can be performed only when RFSHCTL3.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Deep power-down operating modes or Maximum Power Saving Mode.</p> <p>Programming Mode: Dynamic</p> |
| 0 | RW | 0x0 | <p>rank0_refresh</p> <p>Setting this register bit to 1 indicates to the uMCTL2 to issue a refresh to rank 0. Writing to this bit causes DBGSTAT.rank0_refresh_busy to be set. When DBGSTAT.rank0_refresh_busy is cleared, the command has been stored in uMCTL2.</p> <p>This operation can be performed only when RFSHCTL3.dis_auto_refresh=1. It is recommended NOT to set this register bit if in Init or Deep power-down operating modes or Maximum Power Saving Mode.</p> <p>Programming Mode: Dynamic</p> |

DDRC DBGSTAT

Address: Operational Base + offset (0x0310)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:6 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 5 | RO | 0x0 | <p>ctrlupd_busy SoC core may initiate a ctrlupd operation only if this signal is low. This signal goes high in the clock after the uMCTL2 accepts the ctrlupd request. It goes low when the ctrlupd operation is initiated in the uMCTL2. It is recommended not to perform ctrlupd operations when this signal is high.</p> <p>0 - Indicates that the SoC core can initiate a ctrlupd operation 1 - Indicates that ctrlupd operation has not been initiated yet in the uMCTL2</p> <p>Programming Mode: Dynamic</p> |
| 4 | RO | 0x0 | <p>zq_calib_short_busy SoC core may initiate a ZQCS (ZQ calibration short) operation only if this signal is low. This signal goes high in the clock after the uMCTL2 accepts the ZQCS request. It goes low when the ZQCS operation is initiated in the uMCTL2. It is recommended not to perform ZQCS operations when this signal is high.</p> <p>0 - Indicates that the SoC core can initiate a ZQCS operation 1 - Indicates that ZQCS operation has not been initiated yet in the uMCTL2</p> <p>Programming Mode: Dynamic</p> |
| 3:2 | RO | 0x0 | reserved |
| 1 | RO | 0x0 | <p>rank1_refresh_busy SoC core may initiate a rank1_refresh operation (refresh operation to rank 1) only if this signal is low. This signal goes high in the clock after DBGCMD.rank1_refresh is set to one. It goes low when the rank1_refresh operation is stored in the uMCTL2. It is recommended not to perform rank1_refresh operations when this signal is high.</p> <p>0 - Indicates that the SoC core can initiate a rank1_refresh operation 1 - Indicates that rank1_refresh operation has not been stored yet in the uMCTL2</p> <p>Programming Mode: Dynamic</p> |
| 0 | RO | 0x0 | <p>rank0_refresh_busy SoC core may initiate a rank0_refresh operation (refresh operation to rank 0) only if this signal is low. This signal goes high in the clock after DBGCMD.rank0_refresh is set to one. It goes low when the rank0_refresh operation is stored in the uMCTL2. It is recommended not to perform rank0_refresh operations when this signal is high.</p> <p>0 - Indicates that the SoC core can initiate a rank0_refresh operation 1 - Indicates that rank0_refresh operation has not been stored yet in the uMCTL2</p> <p>Programming Mode: Dynamic</p> |

DDRC_SWCTL

Address: Operational Base + offset (0x0320)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | <p>sw_done</p> <p>Enable quasi-dynamic register programming outside reset.</p> <p>Program register to 0 to enable quasi-dynamic programming. Set back register to 1 once programming is done.</p> <p>Programming Mode: Dynamic</p> |

DDRC_SWSTAT

Address: Operational Base + offset (0x0324)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x1 | <p>sw_done_ack</p> <p>Register programming done. This register is the echo of SWCTL.sw_done. Wait for sw_done value 1 to propagate to sw_done_ack at the end of the programming sequence to ensure that the correct registers values are propagated to the destination clock domains.</p> <p>Programming Mode: Static</p> |

DDRC_POISONCFG

Address: Operational Base + offset (0x036c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | <p>rd_poison_intr_clr</p> <p>Interrupt clear for read transaction poisoning. Allow 2/3 clock cycles for correct value to propagate to core logic and clear the interrupts.</p> <p>Programming Mode: Dynamic</p> |
| 23:21 | RO | 0x0 | reserved |
| 20 | RW | 0x1 | <p>rd_poison_intr_en</p> <p>If set to 1, enables interrupts for read transaction poisoning</p> <p>Programming Mode: Dynamic</p> |
| 19:17 | RO | 0x0 | reserved |
| 16 | RW | 0x1 | <p>rd_poison_slverr_en</p> <p>If set to 1, enables SLVERR response for read transaction poisoning</p> <p>Programming Mode: Dynamic</p> |
| 15:9 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 8 | W1C | 0x0 | wr_poison_intr_clr Interrupt clear for write transaction poisoning. Allow 2/3 clock cycles for correct value to propagate to core logic and clear the interrupts. Programming Mode: Dynamic |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x1 | wr_poison_intr_en If set to 1, enables interrupts for write transaction poisoning Programming Mode: Dynamic |
| 3:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | wr_poison_slverr_en If set to 1, enables SLVERR response for write transaction poisoning Programming Mode: Dynamic |

DDRC POISONSTAT

Address: Operational Base + offset (0x0370)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16 | RO | 0x0 | rd_poison_intr_0 Read transaction poisoning error interrupt for port 0. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's read address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register rd_poison_intr_clr, then value propagated to APB clock. Programming Mode: Dynamic |
| 15:1 | RO | 0x0 | reserved |
| 0 | RO | 0x0 | wr_poison_intr_0 Write transaction poisoning error interrupt for port 0. This register is a APB clock copy (double register synchronizer) of the interrupt asserted when a transaction is poisoned on the corresponding AXI port's write address channel. Bit 0 corresponds to Port 0, and so on. Interrupt is cleared by register wr_poison_intr_clr, then value propagated to APB clock. Programming Mode: Dynamic |

DDRC PSTAT

Address: Operational Base + offset (0x03fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16 | RO | 0x0 | wr_port_busy_0 Indicates if there are outstanding writes for AXI port 0. Programming Mode: Dynamic |
| 15:1 | RO | 0x0 | reserved |
| 0 | RO | 0x0 | rd_port_busy_0 Indicates if there are outstanding reads for AXI port 0. Programming Mode: Dynamic |

DDRC_PCCFG

Address: Operational Base + offset (0x0400)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | bl_exp_mode Burst length expansion mode. By default (i.e. bl_exp_mode==0) XPI expands every AXI burst into multiple HIF commands, using the memory burst length as a unit. If set to 1, then XPI will use half of the memory burst length as a unit. This applies to both reads and writes. When MSTR.data_bus_width==00, setting bl_exp_mode to 1 has no effect. This can be used in cases where Partial Writes is enabled (UMCTL2_PARTIAL_WR=1), in order to avoid or minimize t_ccd_l penalty in DDR and t_ccd_mw penalty in LPDDR. Hence, bl_exp_mode=1 is only recommended if DDR or LPDDR. Note that if DBICTL.dm_en=0, functionality is not supported in the following cases: UMCTL2_PARTIAL_WR=0 UMCTL2_PARTIAL_WR=1, MSTR.data_bus_width=01, MEMC_BURST_LENGTH=8 and MSTR.burst_rdwr=1000 (LPDDR only) UMCTL2_PARTIAL_WR=1, MSTR.data_bus_width=01, MEMC_BURST_LENGTH=4 and MSTR.burst_rdwr=0100 (DDR only), with either MSTR.burstchop=0 or CRCPARCTL1.crc_enable=1 Programming Mode: Static |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | pagematch_limit Page match four limit. If set to 1, limits the number of consecutive same page DDRC transactions that can be granted by the Port Arbiter to four when Page Match feature is enabled. If set to 0, there is no limit imposed on number of consecutive same page DDRC transactions. Programming Mode: Static |
| 3:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 0 | RW | 0x0 | <p>go2critical_en</p> <p>If set to 1 (enabled), sets co_gs_go2critical_wr and co_gs_go2critical_lpr / co_gs_go2critical_hpr signals going to DDRC based on urgent input (awurgent, arurgent) coming from AXI master. If set to 0 (disabled), co_gs_go2critical_wr and co_gs_go2critical_lpr / co_gs_go2critical_hpr signals at DDRC are driven to 1b'0.</p> <p>Programming Mode: Static</p> |

DDRC PCFGR 0

Address: Operational Base + offset (0x0404)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:15 | RO | 0x0 | reserved |
| 14 | RW | 0x0 | <p>rd_port_pagematch_en</p> <p>If set to 1, enables the Page Match feature. If enabled, once a requesting port is granted, the port is continued to be granted if the following immediate commands are to the same memory page (same bank and same row). See also related PCCFG.pagematch_limit register.</p> <p>Programming Mode: Static</p> |
| 13 | RW | 0x0 | <p>rd_port_urgent_en</p> <p>If set to 1, enables the AXI urgent sideband signal (arurgent). When enabled and arurgent is asserted by the master, that port becomes the highest priority and co_gs_go2critical_lpr/co_gs_go2critical_hpr signal to DDRC is asserted if enabled in PCCFG.go2critical_en register. Note that arurgent signal can be asserted anytime and as long as required which is independent of address handshaking (it is not associated with any particular command).</p> <p>Programming Mode: Static</p> |
| 12 | RW | 0x0 | <p>rd_port_aging_en</p> <p>If set to 1, enables aging function for the read channel of the port.</p> <p>Programming Mode: Static</p> |
| 11:10 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 9:0 | RW | 0x000 | <p>rd_port_priority</p> <p>Determines the initial load value of read aging counters. These counters will be parallel loaded after reset, or after each grant to the corresponding port. The aging counters down-count every clock cycle where the port is requesting but not granted. The higher significant 5-bits of the read aging counter sets the priority of the read channel of a given port. Port's priority will increase as the higher significant 5-bits of the counter starts to decrease. When the aging counter becomes 0, the corresponding port channel will have the highest priority level (timeout condition -Priority0). For single port configurations, the aging counters are only used when they timeout (become 0) to force read-write direction switching. In this case, external dynamic priority input, arqos (for reads only) can still be used to set the DDRC read priority (2 priority levels: low priority read - LPR, high priority read - HPR) on a command by command basis. Note: The two LSBs of this register field are tied internally to 2'b00.</p> <p>Programming Mode: Static</p> |

DDRC PCFGW_0

Address: Operational Base + offset (0x0408)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:15 | RO | 0x0 | reserved |
| 14 | RW | 0x1 | <p>wr_port_pagematch_en</p> <p>If set to 1, enables the Page Match feature. If enabled, once a requesting port is granted, the port is continued to be granted if the following immediate commands are to the same memory page (same bank and same row). See also related PCCFG.pagematch_limit register.</p> <p>Programming Mode: Static</p> |
| 13 | RW | 0x0 | <p>wr_port_urgent_en</p> <p>If set to 1, enables the AXI urgent sideband signal (awurgent). When enabled and awurgent is asserted by the master, that port becomes the highest priority and co_gs_go2critical_wr signal to DDRC is asserted if enabled in PCCFG.go2critical_en register.</p> <p>Note that awurgent signal can be asserted anytime and as long as required which is independent of address handshaking (it is not associated with any particular command).</p> <p>Programming Mode: Static</p> |
| 12 | RW | 0x0 | <p>wr_port_aging_en</p> <p>If set to 1, enables aging function for the write channel of the port.</p> <p>Programming Mode: Static</p> |
| 11:10 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9:0 | RW | 0x000 | <p>wr_port_priority</p> <p>Determines the initial load value of write aging counters. These counters will be parallel loaded after reset, or after each grant to the corresponding port. The aging counters down-count every clock cycle where the port is requesting but not granted. The higher significant 5-bits of the write aging counter sets the initial priority of the write channel of a given port. Port's priority will increase as the higher significant 5-bits of the counter starts to decrease. When the aging counter becomes 0, the corresponding port channel will have the highest priority level. For single port configurations, the aging counters are only used when they timeout (become 0) to force read-write direction switching. Note: The two LSBs of this register field are tied internally to 2'b00.</p> <p>Programming Mode: Static</p> |

DDRC_PCTRL_0

Address: Operational Base + offset (0x0490)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | <p>port_en</p> <p>Enables AXI port 0.</p> <p>Programming Mode: Dynamic</p> |

DDRPHY_REG0

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x0 | <p>channel select</p> <p>4'b0011: DQ 16bit</p> <p>4'b1111: DQ 32bit</p> |
| 3 | RW | 0x1 | soft reset 1, active low |
| 2 | RW | 0x1 | soft reset 0, active low |
| 1 | RW | 0x1 | |
| 0 | RW | 0x1 | |

DDRPHY_REG1

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | Read ODT bypass mode enable signal, High active. Released register REG03 and REG04 |
| 3 | RW | 0x1 | PHY burst type select: 1: burst8 |
| 2:0 | RW | 0x0 | PHY working condition select 0x0: ddr2 PHY mode 0x1: lpddr2 PHY mode 0x2: ddr3 PHY mode 0x3: lpddr3 PHY mode 0x4: DDR PHY mode |

DDRPHY REG2

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:6 | RW | 0x0 | Write leveling CS select signal 2'b00: select CS0 and CS1 2'b01: select CS1 2'b10: select CS0 |
| 5:4 | RW | 0x0 | DQS gating calibration CS select signal 2'b00: select CS0 and CS1 2'b01: select CS1 2'b10: select CS0 |
| 3 | RW | 0x0 | Write leveling calibration bypass mode, active high |
| 2 | RW | 0x0 | Write leveling calibration control, active high |
| 1 | RW | 0x0 | DQS gating calibration bypass mode, active high |
| 0 | RW | 0x0 | DQS gating calibration control, active high |

DDRPHY REG3

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | reserved |
| 6:4 | RW | 0x2 | right channel A read ODT delay by read odt configure bypass mode |
| 3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x2 | left channel A read ODT delay by read odt configure bypass mode |

DDRPHY REG4

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | reserved |
| 6:4 | RW | 0x2 | right channel B read ODT delay by read odt configure bypass mode |
| 3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x2 | left channel B read ODT delay by read odt configure bypass mode |

DDRPHY REG5

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | Write leveling load mode[7:0] |

DDRPHY REG6

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:6 | RW | 0x0 | Write leveling load mode select[1:0] |
| 5:0 | RW | 0x02 | Write leveling load mode[13:8] |

DDRPHY REG9

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------|
| 31:7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | 0: Normal mode; 1: Bypass |
| 5:0 | RO | 0x0 | reserved |

DDRPHY REGA

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x00 | DDR2 /DDR3/DDR CAS Latency; LPDDR2/3 RL value |

DDRPHY REGB

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | DDR2/DDR3/DDR additive latency |

DDRPHY REGC

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | DDR2/DDR3/DDR WRITE CAS Latency; LPDDR2/3 WL value |

DDRPHY REG11

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | CMD NRCOMP, except for CK/CKB. The larger the value, the stronger the drive strength |

DDRPHY REG12

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:3 | RW | 0x14 | CMD PRCOMP, except for CK/CKB. The larger the value, the stronger the driver strength |
| 2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | CMD weak pull up enable, active low |
| 0 | RW | 0x0 | CMD weak pull down enable, active high |

DDRPHY REG13

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | CMD DLL clock phase select in bypass mode 0: no delay 1: 90°delay |
| 3 | RW | 0x1 | CMD DLL enable 0: disable 1: enable |
| 2:0 | RW | 0x4 | CMD AND ADDRESS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG14

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3 | RW | 0x1 | CK DLL clock phase select in bypass mode 0: no delay 1: 90°delay |
| 2:0 | RW | 0x0 | CK DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG15

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | |

DDRPHY REG16

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | CK/CKB NRCOMP. The larger the value, the stronger the drive strength |

DDRPHY REG17

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | reserved |
| 6:4 | RW | 0x4 | CMD/CK falling edge slew rate control, larger value means larger falling slew rate |
| 3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x4 | CMD/CK rising edge slew rate control, larger vale means larger rising slew rate |

DDRPHY REG18

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | CK/CKB PRCOMP. The larger the value, the stronger the drive strength |

DDRPHY REG1B

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | |

DDRPHY REG1F

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x3 | The response timing when detect the dfi_lp_req is high. Unit: dfi_clk1x |
| 3:0 | RW | 0x0 | The response timing when high the dfi_lp_ack. Unit: dfi_clk1x |

DDRPHY REG20

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Left channel A NRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ0 to A_DQ7 |

DDRPHY REG21

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Left channel A read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from A_DQ0 to A_DQ7 |

DDRPHY REG22

Address: Operational Base + offset (0x0088)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | Left channel A DQ weak pull up enable, active low |
| 0 | RW | 0x0 | Left channel A DQ weak pull down enable, active high |

DDRPHY REG26

Address: Operational Base + offset (0x0098)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | Left channel A write DQ DLL phase select in bypass mode. 0: no delay 1: 90°delay |
| 3 | RW | 0x1 | Left channel A write DQ DLL enable, active HIGH |
| 2:0 | RW | 0x4 | Left channel A write DQ DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG27

Address: Operational Base + offset (0x009c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | Left channel A write DQS DLL phase select in bypass mode. 0: no delay 1: 90°delay |
| 2:0 | RW | 0x0 | Left channel A write DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY_REG28

Address: Operational Base + offset (0x00a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x1 | Left channel A read DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay |

DDRPHY_REG29

Address: Operational Base + offset (0x00a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | Weak pull up of the A_DQS[0] |
| 0 | RW | 0x0 | Weak pull down of the A_DQS[0]. REG29[1:0]=2'b00. Pull Up REG29[1:0]=2'b01. Middle Level REG29[1:0]=2'b10. High-Z REG29[1:0]=2'b11. Pull Down |

DDRPHY_REG2B

Address: Operational Base + offset (0x00ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x4 | Left channel A falling edge slew rate control, larger value means larger falling slew rate |
| 4:2 | RW | 0x0 | Left channel A rising edge slew rate control, larger value means larger rising slew rate |
| 1:0 | RW | 0x1 | |

DDRPHY_REG2C

Address: Operational Base + offset (0x00b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure low 8bit of A channel when in calibartion bypass mode for CS0. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure low 8bit of A channel when in calibartion bypass mode for CS0. Related register REG02[1] |
| 2:0 | RW | 0x0 | Write the dll configure low 8bit of A channel when in calibartion bypass mode for CS0. Related register REG02[1] |

DDRPHY_REG2D

Address: Operational Base + offset (0x00b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure low 8bit of A channel when in calibartion bypass mode for CS1. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure low 8bit of A channel when in calibartion bypass mode for CS1. Related register REG02[1] |
| 2:0 | RW | 0x0 | Write the dll configure low 8bit of A channel when in calibartion bypass mode for CS1. Related register REG02[1] |

DDRPHY_REG2E

Address: Operational Base + offset (0x00b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Left channel A read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from A_DQ0 to A_DQ7 |

DDRPHY_REG2F

Address: Operational Base + offset (0x00bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Left channel A PRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ0 to A_DQ7 |

DDRPHY_REG30

Address: Operational Base + offset (0x00c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Right channel A NRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ8 to A_DQ15 |

DDRPHY_REG31

Address: Operational Base + offset (0x00c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Right channel A read pull-down ODT. The larger the value, the smaller the pull-down resistance in the scope from A_DQ8 to A_DQ15 |

DDRPHY_REG32

Address: Operational Base + offset (0x00c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | Right channel A DQ weak pull up enable, active low |
| 0 | RW | 0x0 | Right channel A DQ weak pull down enable, active high |

DDRPHY REG36

Address: Operational Base + offset (0x00d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | Right channel A write DQ DLL phase select in bypass mode 0: no delay 1: 90°delay |
| 3 | RW | 0x0 | Right channel A write DQ DLL enable, active high |
| 2:0 | RW | 0x4 | Right channel A write DQ DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG37

Address: Operational Base + offset (0x00dc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | Right channel A write DQS DLL phase select in bypass mode. 0: no delay 1: 90°delay |
| 2:0 | RW | 0x0 | Right channel A write DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG38

Address: Operational Base + offset (0x00e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x1 | Right channel A read DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay |

DDRPHY REG39

Address: Operational Base + offset (0x00e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | Weak pull up of the A_DQS[1] |
| 0 | RW | 0x0 | Weak pull down of the A_DQS[1]. REG39[1:0] = 2'b00. Pull Up. REG39[1:0] = 2'b01. Middle level. REG39[1:0] = 2'b10. High-Z. REG39[1:0] = 2'b11. Pull Down |

DDRPHY_REG3A

Address: Operational Base + offset (0x00e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x01 | |

DDRPHY_REG3B

Address: Operational Base + offset (0x00ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x4 | Right channel A falling edge slew rate control, larger value means larger falling slew rate |
| 4:2 | RW | 0x0 | Right channel A rising edge slew rate control, larger value means larger rising slew rate |
| 1:0 | RW | 0x1 | |

DDRPHY_REG3C

Address: Operational Base + offset (0x00f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure high 8bit of A channel when in calibration bypass mode for CS0. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure high 8bit of A channel when in calibration bypass mode for CS0. Related register REG02[1] |
| 2:0 | RW | 0x0 | Write the dll configure high 8bit of A channel when in calibration bypass mode for CS0. Related register REG02[1] |

DDRPHY_REG3D

Address: Operational Base + offset (0x00f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure high 8bit of A channel when in calibration bypass mode for CS1. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure high 8bit of A channel when in calibration bypass mode for CS1. Related register REG02[1] |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 2:0 | RW | 0x0 | Write the dll config high 8bit of A channel when in calibration bypass mode for CS1. Related register REG02[1] |

DDRPHY REG3E

Address: Operational Base + offset (0x00f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Right channel A read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from A_DQ8 to A_DQ15 |

DDRPHY REG3F

Address: Operational Base + offset (0x00fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Right channel A PRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ8 to A_DQ15 |

DDRPHY REG40

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Left channel B NRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ0 to B_DQ7 |

DDRPHY REG41

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Left channel B read pull-down ODT. The larger the value, the smaller the pull-down resistance in the scope from B_DQ0 to B_DQ7 |

DDRPHY REG42

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | Left channel B DQ weak pull up enable, active low |
| 0 | RW | 0x0 | Left channel B DQ weak pull down enable, active high |

DDRPHY REG46

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | Left channel B write DQ DLL phase select in bypass mode. 0: no delay 1: 90°delay |
| 3 | RW | 0x1 | Left channel B write DQ DLL enable, active HIGH |
| 2:0 | RW | 0x4 | Left channel B write DQ DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG47

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | Left channel B write DQS DLL phase select in bypass mode. 0: no delay 1: 90°delay |
| 2:0 | RW | 0x0 | Left channel B write DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG48

Address: Operational Base + offset (0x0120)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x1 | Left channel B read DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay |

DDRPHY REG49

Address: Operational Base + offset (0x0124)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | Weak pull up of the B_DQS[0] |
| 0 | RW | 0x0 | Weak pull down of the B_DQS[0]. REG49[1:0] = 2'b00. Pull Up. REG49[1:0] = 2'b01. Middle level. REG49[1:0] = 2'b10. High-Z. REG49[1:0] = 2'b11. Pull Down |

DDRPHY REG4A

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x01 | |

DDRPHY REG4B

Address: Operational Base + offset (0x012c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x4 | Left channel B falling edge slew rate control, larger value means larger falling slew rate |
| 4:2 | RW | 0x4 | Left channel B rising edge slew rate control, larger value means larger rising slew rate |
| 1:0 | RW | 0x1 | |

DDRPHY REG4C

Address: Operational Base + offset (0x0130)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure low 8bit of B channel when in calibration bypass mode for CS0. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure low 8bit of B channel when in calibration bypass mode for CS0. Related register REG02[1] |
| 2:0 | RW | 0x0 | Write the dll configure low 8bit of B channel when in calibration bypass mode for CS0. Related register REG02[1] |

DDRPHY REG4D

Address: Operational Base + offset (0x0134)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure low 8bit of B channel when in calibration bypass mode for CS1. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure low 8bit of B channel when in calibration bypass mode for CS1. Related register REG02[1] |
| 2:0 | RW | 0x0 | Write the dll configure low 8bit of B channel when in calibration bypass mode for CS1. Related register REG02[1] |

DDRPHY REG4E

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Left channel B read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from B_DQ0 to B_DQ7 |

DDRPHY REG4F

Address: Operational Base + offset (0x013c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Left channel B PRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ0 to B_DQ7 |

DDRPHY REG50

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Right channel B NRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ8 to B_DQ15 |

DDRPHY REG51

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Right channel B read pull-down ODT. The larger the value, the smaller the pull-down resistance in the scope from B_DQ8 to B_DQ15 |

DDRPHY REG52

Address: Operational Base + offset (0x0148)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | Right channel B DQ weak pull up enable, active low |
| 0 | RW | 0x0 | Right channel B DQ weak pull down enable, active high |

DDRPHY REG56

Address: Operational Base + offset (0x0158)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | Right channel B write DQ DLL phase select in bypass mode. 0: no delay 1: 90°delay |
| 3 | RW | 0x1 | Right channel B write DQ DLL enable, active HIGH |
| 2:0 | RW | 0x4 | Right channel B write DQ DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG57

Address: Operational Base + offset (0x015c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | Right channel B write DQS DLL phase select in bypass mode. 0: no delay 1: 90°delay |
| 2:0 | RW | 0x0 | Right channel B write DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay 4: 90°delay 5: 112.5°delay 6: 135°delay 7: 157.5°delay |

DDRPHY REG58

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x1 | Right channel B read DQS DLL delay 0: no delay 1: 22.5°delay 2: 45°delay 3: 67.5°delay |

DDRPHY REG59

Address: Operational Base + offset (0x0164)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | Weak pull up of the B_DQS[1] |
| 0 | RW | 0x0 | Weak pull down of the B_DQS[1]. REG59[1:0] = 2'b00. Pull Up. REG59[1:0] = 2'b01. Middle level. REG59[1:0] = 2'b10. High-Z. REG59[1:0] = 2'b11. Pull Down |

DDRPHY REG5A

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x01 | |

DDRPHY REG5B

Address: Operational Base + offset (0x016c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x4 | Right channel B falling edge slew rate control, larger value means larger falling slew rate |
| 4:2 | RW | 0x4 | Right channel B rising edge slew rate control, larger value means larger rising slew rate |
| 1:0 | RW | 0x1 | |

DDRPHY_REG5C

Address: Operational Base + offset (0x0170)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure high 8bit of B channel when in calibration bypass mode for CS0. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure high 8bit of B channel when in calibration bypass mode for CS0. Related register REG02[1] |
| 2:0 | RW | 0x0 | Write the dll configure high 8bit of B channel when in calibration bypass mode for CS0. Related register REG02[1] |

DDRPHY_REG5D

Address: Operational Base + offset (0x0174)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RW | 0x0 | Write the cyclesel configure high 8bit of B channel when in calibration bypass mode for CS1. Related register REG02[1] |
| 4:3 | RW | 0x0 | Write the ophsel configure high 8bit of B channel when in calibration bypass mode for CS1. Related register REG02[1] |
| 2:0 | RW | 0x0 | Write the dll configure high 8bit of B channel when in calibration bypass mode for CS1. Related register REG02[1] |

DDRPHY_REG5E

Address: Operational Base + offset (0x0178)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x04 | Right channel B read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from B_DQ8 to B_DQ15 |

DDRPHY_REG5F

Address: Operational Base + offset (0x017c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x14 | Right channel B PRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ8 to B_DQ15 |

DDRPHY_REG70

Address: Operational Base + offset (0x01c0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DM0 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DM0 TX de-skew |

DDRPHY_REG71

Address: Operational Base + offset (0x01c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ0 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ0 TX de-skew |

DDRPHY REG72

Address: Operational Base + offset (0x01c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ1 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ1 TX de-skew |

DDRPHY REG73

Address: Operational Base + offset (0x01cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ2 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ2 TX de-skew |

DDRPHY REG74

Address: Operational Base + offset (0x01d0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ3 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ3 TX de-skew |

DDRPHY REG75

Address: Operational Base + offset (0x01d4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ4 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ4 TX de-skew |

DDRPHY REG76

Address: Operational Base + offset (0x01d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ5 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ5 TX de-skew |

DDRPHY REG77

Address: Operational Base + offset (0x01dc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ6 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ6 TX de-skew |

DDRPHY REG78

Address: Operational Base + offset (0x01e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ7 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ7 TX de-skew |

DDRPHY REG79

Address: Operational Base + offset (0x01e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQS0 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQS0 TX de-skew |

DDRPHY REG7A

Address: Operational Base + offset (0x01e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS0 A_DQSB0 TX de-skew |

DDRPHY REG7B

Address: Operational Base + offset (0x01ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DM1 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DM1 TX de-skew |

DDRPHY REG7C

Address: Operational Base + offset (0x01f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ8 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ8 TX de-skew |

DDRPHY REG7D

Address: Operational Base + offset (0x01f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ9 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ9 TX de-skew |

DDRPHY REG7E

Address: Operational Base + offset (0x01f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ10 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ10 TX de-skew |

DDRPHY REG7F

Address: Operational Base + offset (0x01fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ11 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ11 TX de-skew |

DDRPHY REG80

Address: Operational Base + offset (0x0200)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ12 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ12 TX de-skew |

DDRPHY REG81

Address: Operational Base + offset (0x0204)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ13 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ13 TX de-skew |

DDRPHY REG82

Address: Operational Base + offset (0x0208)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQ14 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ14 TX de-skew |

DDRPHY REG83

Address: Operational Base + offset (0x020c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0_A_DQ15 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQ15 TX de-skew |

DDRPHY_REG84

Address: Operational Base + offset (0x0210)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 A_DQS1 RX de-skew |
| 3:0 | RW | 0x7 | CS0 A_DQS1 TX de-skew |

DDRPHY_REG85

Address: Operational Base + offset (0x0214)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS0 A_DQSB1 TX de-skew |

DDRPHY_REG86

Address: Operational Base + offset (0x0218)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DM0 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DM0 TX de-skew |

DDRPHY_REG87

Address: Operational Base + offset (0x021c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ0 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ0 TX de-skew |

DDRPHY_REG88

Address: Operational Base + offset (0x0220)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ1 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ1 TX de-skew |

DDRPHY_REG89

Address: Operational Base + offset (0x0224)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ2 RX de-skew |
| 3:0 | RW | 0x7 | FCS0 B_DQ2 TX de-skew |

DDRPHY_REG8A

Address: Operational Base + offset (0x0228)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ3 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ3 TX de-skew |

DDRPHY REG8B

Address: Operational Base + offset (0x022c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ4 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ4 TX de-skew |

DDRPHY REG8C

Address: Operational Base + offset (0x0230)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ5 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ5 TX de-skew |

DDRPHY REG8D

Address: Operational Base + offset (0x0234)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ6 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ6 TX de-skew |

DDRPHY REG8E

Address: Operational Base + offset (0x0238)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ7 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ7 TX de-skew |

DDRPHY REG8F

Address: Operational Base + offset (0x023c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQS0 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQS0 TX de-skew |

DDRPHY REG90

Address: Operational Base + offset (0x0240)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS0 B_DQSB0 TX de-skew |

DDRPHY REG91

Address: Operational Base + offset (0x0244)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DM1 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DM1 TX de-skew |

DDRPHY REG92

Address: Operational Base + offset (0x0248)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ8 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ8 TX de-skew |

DDRPHY REG93

Address: Operational Base + offset (0x024c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ9 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ9 TX de-skew |

DDRPHY REG94

Address: Operational Base + offset (0x0250)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ10 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ10 TX de-skew |

DDRPHY REG95

Address: Operational Base + offset (0x0254)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ11 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ11 TX de-skew |

DDRPHY REG96

Address: Operational Base + offset (0x0258)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ12 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ12 TX de-skew |

DDRPHY REG97

Address: Operational Base + offset (0x025c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ13 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ13 TX de-skew |

DDRPHY REG98

Address: Operational Base + offset (0x0260)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ14 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ14 TX de-skew |

DDRPHY REG99

Address: Operational Base + offset (0x0264)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQ15 RX de-skew |
| 3:0 | RW | 0x7 | CS0 B_DQ15 TX de-skew |

DDRPHY REG9A

Address: Operational Base + offset (0x0268)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS0 B_DQS1 RX de-skew |
| 3:0 | RW | 0x7 | |

DDRPHY REG9B

Address: Operational Base + offset (0x026c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS0 B_DQSB1 TX de-skew |

DDRPHY REGA3

Address: Operational Base + offset (0x028c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | reserved |
| 5 | RW | 0x1 | Channel A 16bit DQ VREF select 1: internal mode; 0: external mode |
| 4:0 | RW | 0x10 | Channel A 16bit DQ VREF value. VREF value=(reg_value/32)*VDDQ |

DDRPHY REGAE

Address: Operational Base + offset (0x02b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | Channel B 16bit DQ VREF select. 1: internal mode 0: external mode |
| 4:0 | RW | 0x00 | Channel B 16bit DQ VREF value VREF value=(reg_value/32)*VDDQ |

DDRPHY REGB0

Address: Operational Base + offset (0x02c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A1 de-skew |
| 3:0 | RW | 0x7 | A0 de-skew |

DDRPHY REGB1

Address: Operational Base + offset (0x02c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A3 de-skew |
| 3:0 | RW | 0x7 | A2 de-skew |

DDRPHY REGB2

Address: Operational Base + offset (0x02c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A5 de-skew |
| 3:0 | RW | 0x7 | A4 de-skew |

DDRPHY REGB3

Address: Operational Base + offset (0x02cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A7 de-skew |
| 3:0 | RW | 0x7 | A6 de-skew |

DDRPHY REGB4

Address: Operational Base + offset (0x02d0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A9 de-skew |
| 3:0 | RW | 0x7 | A8 de-skew |

DDRPHY REGB5

Address: Operational Base + offset (0x02d4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A11 de-skew |
| 3:0 | RW | 0x7 | A10 de-skew |

DDRPHY REGB6

Address: Operational Base + offset (0x02d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A13 de-skew |
| 3:0 | RW | 0x7 | A12 de-skew |

DDRPHY REGB7

Address: Operational Base + offset (0x02dc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A15 de-skew |
| 3:0 | RW | 0x7 | A14 de-skew |

DDRPHY REGB8

Address: Operational Base + offset (0x02e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | B1 de-skew |
| 3:0 | RW | 0x7 | B0 de-skew |

DDRPHY REGB9

Address: Operational Base + offset (0x02e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | RAS# de-skew |
| 3:0 | RW | 0x7 | B2 de-skew |

DDRPHY REGBA

Address: Operational Base + offset (0x02e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | WE# de-skew |
| 3:0 | RW | 0x7 | CAS# de-skew |

DDRPHY REGBB

Address: Operational Base + offset (0x02ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CKB de-skew |
| 3:0 | RW | 0x7 | CK de-skew |

DDRPHY_REGBC

Address: Operational Base + offset (0x02f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CKE de-skew |
| 3:0 | RW | 0x7 | ODT0 de-skew |

DDRPHY_REGBD

Address: Operational Base + offset (0x02f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CSB0 de-skew |
| 3:0 | RW | 0x7 | RESETN de-skew |

DDRPHY_REGBE

Address: Operational Base + offset (0x02f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CSB1 de-skew |
| 3:0 | RW | 0x7 | ODT1 de-skew |

DDRPHY_REGC0

Address: Operational Base + offset (0x0300)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DM0 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DM0 TX de-skew |

DDRPHY_REGC1

Address: Operational Base + offset (0x0304)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ0 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ0 TX de-skew |

DDRPHY_REGC2

Address: Operational Base + offset (0x0308)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ1 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ1 TX de-skew |

DDRPHY REGC3

Address: Operational Base + offset (0x030c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ2 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ2 TX de-skew |

DDRPHY REGC4

Address: Operational Base + offset (0x0310)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ3 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ3 TX de-skew |

DDRPHY REGC5

Address: Operational Base + offset (0x0314)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ4 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ4 TX de-skew |

DDRPHY REGC6

Address: Operational Base + offset (0x0318)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ5 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ5 TX de-skew |

DDRPHY REGC7

Address: Operational Base + offset (0x031c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ6 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ6 TX de-skew |

DDRPHY REGC8

Address: Operational Base + offset (0x0320)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ7 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ7 TX de-skew |

DDRPHY REGC9

Address: Operational Base + offset (0x0324)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQS0 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQS0 TX de-skew |

DDRPHY REGCA

Address: Operational Base + offset (0x0328)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS1 A_DQSB0 TX de-skew |

DDRPHY REGCB

Address: Operational Base + offset (0x032c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DM1 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DM1 TX de-skew |

DDRPHY REGCC

Address: Operational Base + offset (0x0330)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ8 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ8 TX de-skew |

DDRPHY REGCD

Address: Operational Base + offset (0x0334)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ9 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ9 TX de-skew |

DDRPHY REGCE

Address: Operational Base + offset (0x0338)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ10 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ10 TX de-skew |

DDRPHY REGCF

Address: Operational Base + offset (0x033c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ11 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ11 TX de-skew |

DDRPHY REGDO

Address: Operational Base + offset (0x0340)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ12 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ12 TX de-skew |

DDRPHY REGD1

Address: Operational Base + offset (0x0344)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ13 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ13 TX de-skew |

DDRPHY REGD2

Address: Operational Base + offset (0x0348)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQ14 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ14 TX de-skew |

DDRPHY REGD3

Address: Operational Base + offset (0x034c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | A_DQ15 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQ15 TX de-skew |

DDRPHY REGD4

Address: Operational Base + offset (0x0350)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 A_DQS1 RX de-skew |
| 3:0 | RW | 0x7 | CS1 A_DQS1 TX de-skew |

DDRPHY REGD5

Address: Operational Base + offset (0x0354)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS1 A_DQSB1 TX de-skew |

DDRPHY REGD6

Address: Operational Base + offset (0x0358)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DM0 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DM0 TX de-skew |

DDRPHY REGD7

Address: Operational Base + offset (0x035c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ0 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ0 TX de-skew |

DDRPHY REGD8

Address: Operational Base + offset (0x0360)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ1 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ1 TX de-skew |

DDRPHY REGD9

Address: Operational Base + offset (0x0364)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ2 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ2 TX de-skew |

DDRPHY REGDA

Address: Operational Base + offset (0x0368)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ3 RX de-skew |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 3:0 | RW | 0x7 | CS1 B_DQ3 TX de-skew |

DDRPHY REGDB

Address: Operational Base + offset (0x036c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ4 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ4 TX de-skew |

DDRPHY REGDC

Address: Operational Base + offset (0x0370)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ5 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ5 TX de-skew |

DDRPHY REGDD

Address: Operational Base + offset (0x0374)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ6 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ6 TX de-skew |

DDRPHY REGDE

Address: Operational Base + offset (0x0378)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ7 RX de-skew |
| 3:0 | RW | 0x0 | CS1 B_DQ7 TX de-skew |

DDRPHY REGDF

Address: Operational Base + offset (0x037c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQS0 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQS0 TX de-skew |

DDRPHY REGEO

Address: Operational Base + offset (0x0380)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS1 B_DQSB0 TX de-skew |

DDRPHY REGE1

Address: Operational Base + offset (0x0384)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DM1 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DM1 TX de-skew |

DDRPHY REGE2

Address: Operational Base + offset (0x0388)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ8 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ8 TX de-skew |

DDRPHY REGE3

Address: Operational Base + offset (0x038c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ9 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ9 TX de-skew |

DDRPHY REGE4

Address: Operational Base + offset (0x0390)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ10 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ10 TX de-skew |

DDRPHY REGE5

Address: Operational Base + offset (0x0394)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ11 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ11 TX de-ske |

DDRPHY REGE6

Address: Operational Base + offset (0x0398)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ12 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ12 TX de-skew |

DDRPHY REGE7

Address: Operational Base + offset (0x039c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ13 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ13 TX de-skew |

DDRPHY REGE8

Address: Operational Base + offset (0x03a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ14 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ14 TX de-skew |

DDRPHY REGE9

Address: Operational Base + offset (0x03a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQ15 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQ15 TX de-skew |

DDRPHY REGEA

Address: Operational Base + offset (0x03a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x7 | CS1 B_DQS1 RX de-skew |
| 3:0 | RW | 0x7 | CS1 B_DQS1 TX de-skew |

DDRPHY REGEB

Address: Operational Base + offset (0x03ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x7 | CS1 B_DQSB1 TX de-skew |

DDRPHY REGEC

Address: Operational Base + offset (0x03b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | |

DDRPHY REGF0

Address: Operational Base + offset (0x03c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3 | RO | 0x0 | Channel B High 8bit write leveling done |
| 2 | RO | 0x0 | Channel B Low 8bit write leveling done |
| 1 | RO | 0x0 | Channel A High 8bit write leveling done |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 0 | RO | 0x0 | Channel A Low 8bit write leveling done |

DDRPHY REGF1

Address: Operational Base + offset (0x03c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RO | 0x0 | Channel A High 8bit write leveling dqs value |
| 3:0 | RO | 0x0 | Channel A Low 8bit write leveling dqs value |

DDRPHY REGF2

Address: Operational Base + offset (0x03c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RO | 0x0 | Channel B High 8bit write leveling dqs value |
| 3:0 | RO | 0x0 | Channel B Low 8bit write leveling dqs value |

DDRPHY REGF3

Address: Operational Base + offset (0x03cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RO | 0x0 | Channel A High 8bit write leveling dqs value for CS1 |
| 3:0 | RO | 0x0 | Channel A Low 8bit write leveling dqs value for CS1 |

DDRPHY REGF4

Address: Operational Base + offset (0x03d0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:4 | RO | 0x0 | Channel B High 8bit write leveling dqs value for CS1 |
| 3:0 | RO | 0x0 | Channel B Low 8bit write leveling dqs value for CS1 |

DDRPHY REGFA

Address: Operational Base + offset (0x03e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3 | RO | 0x0 | Channel B High 8bit dqs gate sample dqs value(idqs) (3) |
| 2 | RO | 0x0 | Channel B Low 8bit dqs gate sample dqs value(idqs) (3) |
| 1 | RO | 0x0 | Channel A High 8bit dqs gate sample dqs value(idqs) (3) |
| 0 | RO | 0x0 | Channel A Low 8bit dqs gate sample dqs value(idqs) (3) |

DDRPHY REGFB

Address: Operational Base + offset (0x03ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RO | 0x0 | Calibration get the dll configure channel A low 8bit(3) |
| 4:3 | RO | 0x0 | Calibration get the ophsel configure channel A low 8bit(3) |
| 2:0 | RO | 0x0 | Calibration get the cyclesel configure channel A low 8bit(3) |

DDRPHY REGFC

Address: Operational Base + offset (0x03f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RO | 0x0 | Calibration get the dll configure channel A high 8bit(3) |
| 4:3 | RO | 0x0 | Calibration get the ophsel configure channel A high 8bit(3) |
| 2:0 | RO | 0x0 | Calibration get the cyclesel configure channel A high 8bit(3) |

DDRPHY REGFD

Address: Operational Base + offset (0x03f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RO | 0x0 | Calibration get the dll configure channel B low 8bit(3) |
| 4:3 | RO | 0x0 | Calibration get the ophsel configure channel B low 8bit(3) |
| 2:0 | RO | 0x0 | Calibration get the cyclesel configure channel B low 8bit(3) |

DDRPHY REGFE

Address: Operational Base + offset (0x03f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:5 | RO | 0x0 | Calibration get the dll configure channel B high 8bit(3) |
| 4:3 | RO | 0x0 | Calibration get the ophsel configure channel B high 8bit(3) |
| 2:0 | RO | 0x0 | Calibration get the cyclesel configure channel B high 8bit(3) |

DDRPHY REGFF

Address: Operational Base + offset (0x03fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:4 | RO | 0x0 | reserved |
| 3 | RO | 0x0 | Channel B High 8bit Calibration done |
| 2 | RO | 0x0 | Channel B Low 8bit Calibration done |
| 1 | RO | 0x0 | Channel A High 8bit Calibration done |
| 0 | RO | 0x0 | Channel A Low 8bit Calibration done |

2.4.3 Registers Summary For DDR Standby

| | | | | |
|-------------------------|--------|---|------------|-------------------|
| <u>DDRSTDBY CON0</u> | 0x0000 | W | 0x00000000 | Control Register0 |
| <u>DDRSTDBY CON1</u> | 0x0004 | W | 0x00000000 | Control Register1 |
| <u>DDRSTDBY CON2</u> | 0x0008 | W | 0x00000000 | Control Register2 |
| <u>DDRSTDBY CON3</u> | 0x000c | W | 0x00000800 | Control Register3 |
| <u>DDRSTDBY STATUS0</u> | 0x0010 | W | 0x00000000 | Status Register0 |

2.4.4 Detail Register Description For DDR Standby**DDRSTDBY CON0**

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | idle_th Idle threshold time. Measured by clk_ddr_stdby(General is ddrphy4x/4). When the time of both memory scheduler and uPCTL in idle status exceed idle_th, it will request uPCTL to enter self-refresh. |
| 15:11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | ddr_cmd_plllock_sel Selects the method to determine whether the phy pll is locked after phy pll powers up. 1'b0: the phy pll lock status signal 'ddr_cmd_plllock' is used by DDRSTDBY to determine that the phy pll is locked. 1'b1: the wait threshold time programmed in th_lock is used by DDRSTDBY to determine that the phy pll is locked. |
| 9 | RW | 0x0 | clkouten_dqcmsg_en If DDRSTDBY enters into standby state, the clkouten_dqcmsg_en controls if the phy pll clk output will be disabled or not. 1'b0: disable the clkouten_dqcmsg_if output. 1'b1: enable the clkouten_dqcmsg_if output. |
| 8 | RW | 0x0 | pllpd_dqcmsg_en If DDRSTDBY enters into standby state, the pllpd_dqcmsg_en controls if the phy pll will be powered down or not. 1'b0: disable the pllpd_dqcmsg_if output. 1'b1: enable the pllpd_dqcmsg_if output. |
| 7 | RW | 0x0 | msch_gate_en 1'b0: disable memory scheduler gated 1'b1: enable memory scheduler gated |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 6 | RW | 0x0 | ddrphy1x_gate_en 1'b0: disable ddrphy1x gated 1'b1: enable ddrphy1x gated |
| 5 | RW | 0x0 | upctl_core_clk_gate_en 1'b0: disable uPCTL core_clk gated 1'b1: enable uPCTL core_clk gated |
| 4 | RW | 0x0 | upctl_aclk_gate_en 1'b0: disable uPCTL aclk gated 1'b1: enable uPCTL aclk gated |
| 3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | sysack_ext_dis 1'b0: exit stdby need to wait assertion of sysack 1'b1: exit stdby not need to wait assertion of sysack |
| 1 | RW | 0x0 | ctl_idle_en 1'b0: disable uPCTL idle when DDRSTDBY is enabled 1'b1: enable uPCTL idle when DDRSTDBY is enabled |
| 0 | RW | 0x0 | stdby_en 1'b0: disable DDRSTDBY function 1'b1: enable DDRSTDBY function |

DDRSTDBY CON1

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RW | 0x0000 | cg_exit_th Clk gated exit threshold time. Measured by clk_ddr_stdby(General is ddrphy4x/4). If ddrphy1x_gate_en=1, cg_exit_th need consider the dll lock time of ddr phy. |
| 15:0 | RW | 0x0000 | cg_wait_th Clk gated wait threshold time. Measured by clk_ddr_stdby(General is ddrphy4x/4). Set this value to 0x0. |

DDRSTDBY CON2

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x00 | th_pdpllen The wait threshold time from the assertion of pllpd_qdcmd_if to the assertion of phy_stby_gate which will gate the ddrphy1x clk of the phy. It takes effect only when pllpd_dqcmsg_en is enabled. |
| 23:16 | RW | 0x00 | th_clkouten The wait threshold time from the assertion of clkouten_qdcmd_if to the assertion of phy_stby_gate which will gate the ddrphy1x clk of the phy. It takes effect only when clkouten_dqcmsg_en is enabled. |
| 15:0 | RW | 0x0000 | th_pdpll The threshold time to make sure that the phy pll should be in power down state. Measured by clk_ddr_stby(General is ddrphy4x/4). It takes effect only when pllpd_dqcmsg_en is enabled. |

DDRSTDBY CON3

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000800 | th_lock The wait threshold time from the deassertion of phy pll power down enable to when the phy pll is regarded as locked. |

DDRSTDBY STATUS0

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:18 | RO | 0x0 | reserved |
| 17 | RW | 0x0 | ddr_cmd_plllock The lock status of the phy pll. |
| 16 | RW | 0x0 | clkouten_dqcmsg_if 1'b0: indicates the phy pll clk output should be gated. 1'b1: indicates the phy pll clk output should not be gated. |
| 15 | RW | 0x0 | pllpd_dqcmsg_if 1'b0: indicates the phy pll should not power down. 1'b1: indicates the phy pll should power down. |
| 14 | RW | 0x0 | phy_stby_gate The phy_stby_gate gates ddrphy1x. 1'b0: DDRSTDBY global gated is disabled 1'b1: DDRSTDBY global gated is enabled |
| 13 | RW | 0x0 | sysactive 1'b0: uPCTL is idle 1'b1: uPCTL is active |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12 | RO | 0x0 | sysack 1'b0: ack the request of sysreq enter self-refresh 1'b1: ack the request of sysreq exit self-refresh |
| 11 | RO | 0x0 | sysreq 1'b0: request uPCTL enter self-refresh 1'b1: request uPCTL exit self-refresh |
| 10 | RO | 0x0 | upctl_idle 1'b0: uPCTL is active 1'b1: uPCTL is idle |
| 9 | RW | 0x0 | pwract 1'b0: memory scheduler is not active 1'b1: memory scheduler is active |
| 8 | RW | 0x0 | ddr_stby_gate The ddr_stby_gate gates msch/upctl2 aclk/core clk. 1'b0: DDRSTDBY global gated is disabled 1'b1: DDRSTDBY global gated is enabled |
| 7 | RO | 0x0 | stdby_en 1'b0: DDRSTDBY is disabled 1'b1: DDRSTDBY is enabled |
| 6:0 | RW | 0x00 | state 4'b0000: ST_NORMAL 4'b0001: ST_STDBY_WAIT_EN 4'b0010: ST_STDBY_SR_ENTRY 4'b0011: ST_STDBY_WAIT 4'b0100: ST_STDBY 4'b0101: ST_STDBY0 4'b0110: ST_STDBY1 4'b0111: ST_PDPLL 4'b1000: ST_WAIT_LOCK 4'b1001: ST_STDBY2 4'b1010: ST_STDBY_SR_EXIT |

2.4.5 Registers Summary For DDR Monitor

| Name | Offset | Size | Reset Value | Description |
|---------------------|---------------|-------------|--------------------|--|
| DDRMON_IP VERSION | 0x0000 | W | 0x00000022 | DDR Monitor IP Version |
| DDRMON_CTRL | 0x0004 | W | 0x00000008 | DDR Monitor Control Register |
| DDRMON_INT_STATUS | 0x0008 | W | 0x00000000 | Interrupt Status |
| DDRMON_INT_MASK | 0x000c | W | 0x00000000 | Interrupt mask control |
| DDRMON_TIMER_COUNT | 0x0010 | W | 0x00000000 | The DFI Timer Threshold |
| DDRMON_FLOOR_NUMBER | 0x0014 | W | 0x00000000 | The Low Threshold in the Comparison of DDR Access |
| DDRMON_TOP_NUMBER | 0x0018 | W | 0x00000000 | The High Threshold in the Comparison of DDR Access |
| DDRMON_DFI_ACT_NUM | 0x001c | W | 0x00000000 | DFI Active Command Number |

| Name | Offset | Size | Reset Value | Description |
|--------------------------|--------|------|-------------|---|
| DDRMON DFI WR NUM | 0x0020 | W | 0x00000000 | DFI Write Command Number |
| DDRMON DFI RD NUM | 0x0024 | W | 0x00000000 | DFI Read Command Number |
| DDRMON COUNT NUM | 0x0028 | W | 0x00000000 | Timer Count Number |
| DDRMON DFI ACCESS NUMBER | 0x002c | W | 0x00000000 | DFI Read And Write Command Number |
| DDRMON TOP LP NUMBER | 0x0030 | W | 0x00000000 | The High Threshold In The Comparison Of DDR Cke Low |
| DDRMON FLOOR LP NUMBER | 0x0034 | W | 0x00000000 | The Low Threshold In The Comparison Of DDR Cke Low |
| DDRMON DFI SREX NUM | 0x0038 | W | 0x00000000 | Number Of Cke Low For DFI Self-refresh |
| DDRMON DFI PDEX NUM | 0x003c | W | 0x00000000 | Number Of Cke Low For DFI Power Down |
| DDRMON DFI CLKSTOP NUM | 0x0040 | W | 0x00000000 | Number Of Cke Low For DFI Clkstop |
| DDRMON DFI LP NUM | 0x0044 | W | 0x00000000 | Total Number Of Cke Low |
| DDRMON DFI PHY LP NUMBER | 0x0048 | W | 0x00000000 | DDR Phy Low Power |
| DDRMON DDR IF CTRL | 0x0200 | W | 0x00000000 | DDR Interface Control Register |
| DDRMON DDR MSTID | 0x0204 | W | 0x00000000 | Master And AXI ID Of DDR Command |
| DDRMON DDR IDMSK | 0x0208 | W | 0x00000000 | Master And AXI ID MASK Of DDR Command |
| DDRMON WR START ADDR | 0x020c | W | 0x00000000 | Write Start Address |
| DDRMON WR END ADDR | 0x0210 | W | 0x00000000 | Write End Address |
| DDRMON RD START ADDR | 0x0214 | W | 0x00000000 | Read Start Address |
| DDRMON RD END ADDR | 0x0218 | W | 0x00000000 | Read End Address |
| DDRMON DDR FIFO0 ADDR | 0x0240 | W | 0x00000000 | DDR Controller Interface Address FIFO0 |
| DDRMON DDR FIFO0 ID | 0x0244 | W | 0x00000000 | DDR Controller Interface Command ID FIFO0 |
| DDRMON DDR FIFO1 ADDR | 0x0248 | W | 0x00000000 | DDR Controller Interface Address FIFO1 |
| DDRMON DDR FIFO1 ID | 0x024c | W | 0x00000000 | DDR Controller Interface Command ID FIFO1 |
| DDRMON DDR FIFO2 ADDR | 0x0250 | W | 0x00000000 | DDR Controller Interface Address FIFO2 |
| DDRMON DDR FIFO2 ID | 0x0254 | W | 0x00000000 | DDR Controller Interface Command ID FIFO2 |
| DDRMON DDR FIFO3 ADDR | 0x0258 | W | 0x00000000 | DDR Controller Interface Address FIFO3 |

| Name | Offset | Size | Reset Value | Description |
|---------------------|--------|------|-------------|---|
| DDRMON DDR FIFO3 ID | 0x025c | W | 0x00000000 | DDR Controller Interface Command ID FIFO3 |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

2.4.6 Detail Register Description For DDR Monitor

DDRMON IP VERSION

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--------------------------------------|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RO | 0x22 | ip_version DDR monitor IP version |

DDRMON CTRL

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | write_enable When bit 16=1, bit 0 can be written by software . When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software. When bit 17=0, bit 1 cannot be written by software; When bit 31=1, bit 15 can be written by software. When bit 31=0, bit 15 cannot be written by software; |
| 15:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | ddr4_en 1'b1: enable 1'b0: disable |
| 4 | RW | 0x0 | lpddr4_en 1'b1: enable 1'b0: disable |
| 3 | RW | 0x1 | hardware_en 1'b1: enable 1'b0: disable |
| 2 | RW | 0x0 | lpddr23_en Enable lpddr2 or lpddr3 1'b1: enable 1'b0: disable |
| 1 | RW | 0x0 | software_en 1'b1: enable 1'b0: disable |
| 0 | RW | 0x0 | timer_cnt_en 1'b1: enable 1'b0: disable |

DDRMON INT STATUS

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | reserved |
| 12 | RW | 0x0 | rd_addr_hit This is the interrupt status read address hit the setting range. |
| 11 | RW | 0x0 | wr_addr_hit This is the interrupt status write address hit the setting range. |
| 10:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | over_int_clkstop This is the interrupt status of DDR clk stop number over than high threshold. Only valid for lpddr2/lpddr3/lpddr4. |
| 7 | RW | 0x0 | below_int_clkstop This is the interrupt status of DDR clk stop number less than low threshold. Only valid for lpddr2/lpddr3/lpddr4. |
| 6 | RO | 0x0 | over_int_pdex This is the interrupt status of DDR power down number over than high threshold. |
| 5 | RO | 0x0 | below_int_pdex This is the interrupt status of DDR power down number less than low threshold. |
| 4 | RW | 0x0 | compare_statistics This is the interrupt status to statistics the number of activate, write or read command and so on. |
| 3 | RW | 0x0 | over_int_srex This is the interrupt status of DDR self refresh number over than high threshold. |
| 2 | RW | 0x0 | below_int_srex This is the interrupt status of DDR self refresh number less than low threshold. |
| 1 | RO | 0x0 | over_int This is the interrupt status of DDR read and write burst number more than high threshold. |
| 0 | RO | 0x0 | below_int This is the interrupt status of DDR read and write burst number less than low threshold. |

DDRMON INT MASK

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:27 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 26:16 | RW | 0x000 | write_enable When bit 16=1, bit 0 can be written by software . When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software. When bit 17=0, bit 1 cannot be written by software; When bit 31=1, bit 15 can be written by software. When bit 31=0, bit 15 cannot be written by software; |
| 15:11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | rd_addr_hit_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 9 | RW | 0x0 | wr_addr_hit_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 8 | RW | 0x0 | over_int_clkstop_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 7 | RW | 0x0 | below_int_clkstop_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 6 | RO | 0x0 | over_int_pdex_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 5 | RO | 0x0 | below_int_pdex_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 4 | RW | 0x0 | compare_statistics_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 3 | RW | 0x0 | over_int_srex_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 2 | RW | 0x0 | below_int_srex_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 1 | RO | 0x0 | over_int_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |
| 0 | RO | 0x0 | below_int_mask Interrupt mask control, when bit set to 1'b1, the corresponding interrupt will disable. |

DDRMON TIMER COUNT

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | timer_count The DFI timer threshold, the statistics of DDR access only be done when timer counter is less than this threshold in hardware mode |

DDRMON FLOOR NUMBER

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | floor_number The low threshold in the comparison of DDR access |

DDRMON TOP NUMBER

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | top_number The high threshold in the comparison of DDR access |

DDRMON DFI ACT NUM

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | dfi_act_num DFI active command number in the statistics range |

DDRMON DFI WR NUM

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | dfi_wr_num DFI write command number in the statistics range |

DDRMON DFI RD NUM

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | dfi_rd_num DFI read command number in the statistics range |

DDRMON COUNT NUM

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | dfi_count_num Timer count number in the statistics range |

DDRMON DFI ACCESS NUM

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | dfi_access_num DFI read and write command number in the statistics range |

DDRMON TOP LP NUMBER

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | top_lp_number The high threshold in the comparison of DDR self-refresh, power down or clkstop. |

DDRMON FLOOR LP NUMBER

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | floor_lp_number The low threshold in the comparison of DDR self-refresh, power down or clkstop. |

DDRMON DFI SREX NUM

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | dfi_srex_num DFI self-refresh number during cke low. |

DDRMON DFI PDEX NUM

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | dfi_pdex_num DFI power down number during cke low. |

DDRMON DFI CLKSTOP NUM

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | dfi_clkstop_num DFI clkstop number during cke low. Only valid for lpddr2/3/4. |

DDRMON DFI LP NUM

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | dfi_lp_num The sum of DFI self-refresh, power down and clkstop number during cke low. |

DDRMON DFI PHY LP NUM

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | dfi_phy_lp_num Phy low power count number during the time both dfi_lp_req and dfi_lp_ack assert. |

DDRMON DDR IF CTRL

Address: Operational Base + offset (0x0200)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | write_enable When bit 16=1, bit 0 can be written by software. When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software. When bit 17=0, bit 1 cannot be written by software; When bit 31=1, bit 15 can be written by software. When bit 31=0, bit 15 cannot be written by software; |
| 15:3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | if_mon_en 1'b1: enable 1'b0: disable |
| 1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | direction 1'b1: read 1'b0: write |

DDRMON DDR MSTID

Address: Operational Base + offset (0x0204)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16:0 | RW | 0x00000 | ddr_mstid High 7 bits: Master ID Low 10 bits: AXI command ID |

DDRMON DDR IDMSK

Address: Operational Base + offset (0x0208)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16:0 | RW | 0x00000 | ddr_idmsk When bit set to high, this bit of MSTID will be masked, and does not take part in the ID comparison |

DDRMON WR START ADDR

Address: Operational Base + offset (0x020c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | wr_start_addr Write start address for address comparison |

DDRMON WR END ADDR

Address: Operational Base + offset (0x0210)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | wr_end_addr Write end address for address comparison |

DDRMON RD START ADDR

Address: Operational Base + offset (0x0214)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | rd_start_addr Read start address for address comparison |

DDRMON RD END ADDR

Address: Operational Base + offset (0x0218)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | rd_end_addr Read end address for address comparison |

DDRMON DDR FIFO0 ADDR

Address: Operational Base + offset (0x0240)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | ddr_fifo0_addr DDR controller interface address FIFO0 |

DDRMON DDR FIFO0 ID

Address: Operational Base + offset (0x0244)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16:0 | RO | 0x00000 | ddr_fifo0_id DDR controller interface command ID FIFO0 |

DDRMON DDR FIFO1 ADDR

Address: Operational Base + offset (0x0248)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | ddr_fifo1_addr DDR controller interface address FIFO1 |

DDRMON DDR FIFO1 ID

Address: Operational Base + offset (0x024c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16:0 | RO | 0x00000 | ddr_fifo1_id DDR controller interface command ID FIFO1 |

DDRMON DDR FIFO2 ADDR

Address: Operational Base + offset (0x0250)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | ddr_fifo2_addr DDR controller interface address FIFO2 |

DDRMON DDR FIFO2 ID

Address: Operational Base + offset (0x0254)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16:0 | RO | 0x00000 | ddr_fifo2_id DDR controller interface command ID FIFO2 |

DDRMON DDR FIFO3 ADDR

Address: Operational Base + offset (0x0258)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | ddr_fifo3_addr DDR controller interface address FIFO3 |

DDRMON DDR FIFO3 ID

Address: Operational Base + offset (0x025c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16:0 | RO | 0x00000 | ddr_fifo3_id DDR controller interface command ID FIFO3 |

2.5 Interface Description

DDR IOs are listed as following Table.

Table 2-1 DDR IO description

| Pin Name | Description |
|--------------------|--|
| CK | Positive differential clock |
| CKB | Negative differential clock |
| CKE | Active-high clock enable signal for two chip select. |
| CSBi (i=0,1) | Active-low chip select signal. There are two chip select. |
| ACTN | Activation command input. For DDR3, this pin should connect to the RASN. for DDR, this pin should connect to the ACTN |
| BA[1:0] BG[1:0] | Bank address signal. For DDR3, BA0 should connect to the BA0. BA1 should connect to BA1. BG0 should connect to BA2. For DDR, please connect directly. Unused for LPDDRn. |

| | |
|--------------|---|
| A[17:0] | Address signal. For DDR3/DDR2, A17 should connect to CASN and A16 should connect to WEN. For DDR, please connect directly. For LPDDRn, only use A9-A0. |
| DQ[31:0] | Bidirectional data line |
| DQS[3:0] | Positive differential bidirectional data strobes |
| DQSB[3:0] | Negative differential bidirectional data strobes. |
| DM[3:0] | Active-low data mask signal. |
| ODTi (i=0,1) | On-Die Termination output signal for two chip select. |
| RESETN | Reset signal. |

The DDR Interface of DDR PHY is reused with DDR3/DDR3L, so the mapping relation between the DDR and DDR3/DDR3L show in the following table

Table 2-2 DDR IO Mapping Table

| PHY TOP-LEVEL PINS | DDR | DDR3 | DDR2 | LPDDR2/3 |
|--------------------|--------|--------|------|----------|
| A0 | A0 | A9 | A0 | A0 |
| A1 | A1 | A14 | A1 | A1 |
| A2 | A2 | A13 | A2 | A2 |
| A3 | A3 | A11 | A3 | A3 |
| A4 | A4 | A2 | A4 | A4 |
| A5 | A5 | A4 | A5 | A5 |
| A6 | A6 | A3 | A6 | A6 |
| A7 | A7 | A6 | A7 | A7 |
| A8 | A8 | A5 | A8 | A8 |
| A9 | A9 | A1 | A9 | A9 |
| A10 | A10 | A0 | A10 | - |
| A11 | A11 | A7 | A11 | - |
| A12 | A12 | CASB | A12 | - |
| A13 | A13 | A8 | A13 | - |
| A14 | A14 | ODT0 | A14 | - |
| A15 | A15 | BA1 | A15 | - |
| A16 | A16 | RASB | WEB | - |
| A17 | A17 | - | CASB | - |
| ACTN | ACTN | CSB0 | RASB | - |
| BA0 | BA0 | BA2 | BA0 | - |
| BA1 | BA1 | A12 | BA1 | - |
| BG0 | BG0 | BA0 | BA2 | - |
| BG1 | BG1 | WEB | - | - |
| CK | CK | CK | CK | CK |
| CKB | CKB | CKB | CKB | CKB |
| CKE | CKE | CKE | CKE | CKE |
| CSB0 | CSB0 | A10 | CSB0 | CSB0 |
| CSB1 | CSB1 | CSB1 | CSB1 | CSB1 |
| ODT0 | ODT0 | A15 | ODT0 | ODT0 |
| ODT1 | ODT1 | ODT1 | ODT1 | ODT1 |
| RESETN | RESETN | RESETN | - | - |

2.6 Application Notes

2.6.1 Initialization

PCTL & DDR PHY Initialization

1. Assert the resets (presetsn, core_ddrc_rstn and aresetn_0 of PCTL)
2. Configure PLL of DDR PHY and wait PLL lock.
3. De-assert presetsn if clocks are active and stable.
4. Initial PCTL register and PHY Register

5. Start PHY initialization with DFIMIS[5] register of PCTL
6. De-assert the remaining resets (core_ddrc_rstn and aresetn_0 of PCTL)
7. Wait PCTL initialization done (STAT.operation_mode==normal)
8. Start PHY dqs calibration with PHYREG02 register and wait calibration finish with PHYREGFF register.
9. (optional) After dqs calibration, start write leveling training with PHYREG02 register and wait write leveling training finish with PHYJREGF0 register.
10. Start write and read

DDR3/DDR3L Initialization Sequence

The initialization steps for DDR3/DDR3L SDRAMs are as follows:

1. Power-up.
2. Maintain dfi_reset_n low for duration specified by INIT1.dram_rstn_x1024. Specification requires at least 200 us with stable power.
3. Issue NOP/deselect for duration specified by INIT0.pre_cke_x1024. Specification requires at least 500 us.
4. Assert CKE and issue NOP/deselect for INIT0.post_cke_x1024 (specification requires at least tXPR).
5. Issue MRS (mode register set) command to load MR2 with INIT4.emr2 value followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
6. Issue MRS command to load MR3 with INIT4.emr3 followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
7. Issue MRS command to load MR1 with INIT4.emr followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
8. Issue MRS command to load MR0 with INIT3.mr followed by NOP/deselect for duration of DRAMTMG3.t_mod.
9. Issue ZQCL command to start ZQ calibration and wait for INIT5.dev_zqinit_x32.
10. Wait for INIT5.dev_zqinit_x32 counting to finish. Ensure wait from step 8 is larger than tDLLK.
11. The PCTL controller is now ready for normal operation.

DDR Initialization Sequence

The initialization steps for DDR SDRAMs are as follows:

1. Power-up.
2. Maintain dfi_reset_n low for duration specified by INIT1.dram_rstn_x1024. Specification requires at least 200 us with stable power.
3. Issue NOP/deselect for duration specified by INIT0.pre_cke_x1024. Specification requires at least 500 us.
4. Assert CKE and issue NOP/deselect for INIT0.post_cke_x1024 (specification requires at least tXPR).
5. Issue MRS (mode register set) command to load MR3 with INIT4.emr3 value followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
6. Issue MRS (mode register set) command to load MR6 with INIT7.mr6 value followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
7. Issue MRS (mode register set) command to load MR5 with INIT6.mr5 value followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
8. Issue MRS (mode register set) command to load MR4 with INIT7.mr4 value followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
9. Issue MRS command to load MR2 with INIT4.emr2 followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
10. Issue MRS command to load MR1 with INIT4.emr followed by NOP/deselect for duration of DRAMTMG3.t_mrd.
11. Issue MRS command to load MR0 with INIT3.mr followed by NOP/deselect for duration of DRAMTMG3.t_mod.
12. Issue ZQCL command to start ZQ calibration and wait for INIT5.dev_zqinit_x32.
13. Wait for INIT5.dev_zqinit_x32 counting to finish. Ensure wait from step 8 is larger than tDLLK.

14. The PCTL controller is now ready for normal operation.

LPDDR2/3 Initialization Sequence

The initialization steps for LPDDR2/3 SDRAMs are as follows:

1. Power-up.
2. CKE is held low for a duration specified by INIT0.pre_cke_x1024. The clock is checked to be stable for duration specified by INIT2.min_stable_clock_x1 (minimum of 5 clock cycles) prior to the first low to high transition of CKE.
3. Assert CKE for INIT0.post_cke_x1024 (specification requires at least 200 us).
4. A MRW (Reset) command is issued to MRW63 register. Values of MA<7:0> = 3FH and OP<7:0> = 00H is used for this command. The MRW reset command brings the device to the device auto initialization (resetting) state in the power-on initialization sequence.
5. Issue NOP/deselect for duration specified by INIT2.idle_after_reset_x32 (specification requires 1 us minimum) and INIT5.max_auto_init_x1024 (specification requires maximum time of 10 us).
6. An MRW ZQ initialization calibration command is issued to the memory to register MR10 to initiate the ZQ calibration. Values of MA<7:0> = 0AH and OP<7:0> = FFH is used for this command.
7. Issue NOP/deselect for duration specified by INIT5.dev_zqinit_x32 (specification requires a minimum time of 1us).
8. Program MR2 register by setting MR2 register to INIT3.emr followed by a NOP/deselect for a duration specified by DRAMTMG3.t_mrw (typical value of 5 clock cycles).
9. Program MR1 register by setting MR1 register to INIT3.mr followed by a NOP/deselect for a duration specified by DRAMTMG3.t_mrw (typical value of 5 clock cycles).
10. Program MR3 register by setting MR3 register to INIT4.emr2 followed by a NOP/deselect for a duration specified by DRAMTMG3.t_mrw (typical value of 5 clock cycles).
11. Schedule multiple all bank refresh.
12. The PCTL controller is now ready for normal operation.

2.6.2 High Speed IO Drive Strength

The tuning range of driver resistance is 19.6ohm to ∞ . By default, 0x14 is 37.6ohm for DDR3 DQ and CMD driver. When the control bit is set to be larger, the drive strength becomes stronger.

Table 2-3 CK/CMD Driver output resistance

| Offset | Bit | Default | Description |
|--------|-----|---------|-------------------------------------|
| 0xd0 | 4~0 | 0xe | adjustable CMD pull-down resistance |
| 0xd4 | 4~0 | 0xe | adjustable CMD pull-up resistance |
| 0xd8 | 4~0 | 0xe | adjustable CK pull-down resistance |
| 0xdc | 4~0 | 0xe | adjustable CK pull-up resistance |

Table 2-4 DM, DQ Signal Drive Strength Register

| Offset | Bit | Default | Description |
|--------|-----|---------|---|
| 0xe0 | 4~0 | 0x8 | pull-down ODT driving resistance for A_DQ0~A_DQ7 |
| 0xe4 | 4~0 | 0x8 | pull-up ODT driving resistance for A_DQ0~A_DQ7 |
| 0xe8 | 4~0 | 0xe | pull-down driving resistance for A_DQ0~A_DQ7 |
| 0xec | 4~0 | 0xe | pull-up driving resistance for A_DQ0~A_D7 |
| 0x100 | 4~0 | 0x8 | pull-down ODT driving resistance for A_DQ8~A_DQ15 |
| 0x104 | 4~0 | 0x8 | pull-up ODT driving resistance for A_DQ8~A_DQ15 |
| 0x108 | 4~0 | 0xe | pull-down driving resistance for A_DQ8~A_DQ15 |
| 0x10c | 4~0 | 0xe | pull-up driving resistance for A_DQ8~A_D15 |
| 0x120 | 4~0 | 0x8 | pull-down ODT driving resistance for B_DQ0~B_DQ7 |
| 0x124 | 4~0 | 0x8 | pull-up ODT driving resistance for B_DQ0~B_DQ7 |
| 0x128 | 4~0 | 0xe | pull-down driving resistance for B_DQ0~B_DQ7 |
| 0x12c | 4~0 | 0xe | pull-up driving resistance for B_DQ0~B_D7 |

| | | | |
|-------|-----|-----|---|
| 0x140 | 4~0 | 0x8 | pull-down ODT driving resistance for B_DQ8~B_DQ15 |
| 0x144 | 4~0 | 0x8 | pull-up ODT driving resistance for B_DQ8~B_DQ15 |
| 0x148 | 4~0 | 0xe | pull-down driving resistance for B_DQ8~B_DQ15 |
| 0x14c | 4~0 | 0xe | pull-up driving resistance for B_DQ8~B_D15 |

The value is larger, the drive strength is stronger.

DDR3 1.5V DQ/DQS/CMD driver and ODT strength table

Table 2-5 DDR3 1.5V DQ/DQS/CMD Driver and ODT resistance

| Control bit | 5'b00000 | 5'b00001 | 5'b00010 | 5'b00011 |
|--------------|----------|----------|----------|----------|
| Pull-up/down | +∞ | 506.1 | 253.1 | 168.7 |
| Control bit | 5'b00100 | 5'b00101 | 5'b00110 | 5'b00111 |
| Pull-up/down | 126.5 | 101.2 | 84.36 | 72.31 |
| Control bit | 5'b01000 | 5'b01001 | 5'b01010 | 5'b01011 |
| Pull-up/down | 63.27 | 56.24 | 50.62 | 46.02 |
| Control bit | 5'b01100 | 5'b01101 | 5'b01110 | 5'b01111 |
| Pull-up/down | 42.18 | 38.94 | 36.15 | 33.74 |
| Control bit | 5'b10000 | 5'b10001 | 5'b10010 | 5'b10011 |
| Pull-up/down | 63.27 | 56.24 | 50.62 | 46.02 |
| Control bit | 5'b10100 | 5'b10101 | 5'b10110 | 5'b10111 |
| Pull-up/down | 42.18 | 38.94 | 36.15 | 33.74 |
| Control bit | 5'b11000 | 5'b11001 | 5'b11010 | 5'b11011 |
| Pull-up/down | 31.64 | 29.77 | 28.12 | 26.64 |
| Control bit | 5'b11100 | 5'b11101 | 5'b11110 | 5'b11111 |
| Pull-up/down | 25.31 | 24.1 | 23.01 | 22.01 |

Table 2-6 DDR/LPDDR2/3 1.2V DQ/DQS/CMD driver and ODT resistance

| Control bit | 5'b00000 | 5'b00001 | 5'b00010 | 5'b00011 |
|--------------|----------|----------|----------|----------|
| Pull-up/down | +∞ | 569.8ohm | 284.9ohm | 189.9ohm |
| Control bit | 5'b00100 | 5'b00101 | 5'b00110 | 5'b00111 |
| Pull-up/down | 142.4 | 114 | 94.97 | 81.4 |
| Control bit | 5'b01000 | 5'b01001 | 5'b01010 | 5'b01011 |
| Pull-up/down | 71.23 | 63.31 | 56.98 | 51.8 |
| Control bit | 5'b01100 | 5'b01101 | 5'b01110 | 5'b01111 |
| Pull-up/down | 47.48 | 43.83 | 40.7 | 37.99 |
| Control bit | 5'b10000 | 5'b10001 | 5'b10010 | 5'b10011 |
| Pull-up/down | 71.23 | 63.31 | 56.98 | 51.8 |
| Control bit | 5'b10100 | 5'b10101 | 5'b10110 | 5'b10111 |
| Pull-up/down | 47.48 | 43.83 | 40.7 | 37.99 |
| Control bit | 5'b11000 | 5'b11001 | 5'b11010 | 5'b11011 |
| Pull-up/down | 35.61 | 33.52 | 31.66 | 29.99 |
| Control bit | 5'b11100 | 5'b11101 | 5'b11110 | 5'b11111 |
| Pull-up/down | 28.49 | 27.13 | 25.9 | 24.77 |

2.6.3 Per bit de-skew tuning

Per-bit de-skew is designed for compensating PCB trace mismatch, DDR PHY support skew individually adjustable for all PHY signals. There are eight steps for each bit de-skew adjusting, and the adjust resolution under different corners is shown below:

Table 2-7 Per-bit de-skew tuning resolution

| | ff | tt | ss |
|--------------------|------|------|------|
| de-skew resolution | 12ps | 15ps | 22ps |

Pre-bit de-skew is realized with inverter chain delay, per-bit de-skew control signals select how much inverters are connected to data path, the minimum resolution is determined by

the two inverters minimum delay.

TX path de-skew and RX path de-skew employ same delay line, and they have same de-skew tuning resolution. Minimum RX de-skew tuning resolution can be about 28ps with SMIC55II tt corner process, and we can re-design tuning resolution according to system and customer requirement.

2.6.4 DDR PHY Calibration

DDR PHY auto dqs calibration function has been implemented in the PHY. The entire training processes only need to configure the register to start and wait for finish.

The entire training process is as follows:

1. PHY's register is reset, the setup is complete.
2. Send the initial command to dram and complete dram initialization.
3. Set the PHY's register beginning calibration.

| Offset | Bit | Default | Description |
|--------|-----|---------|--|
| 0x8 | 5~4 | 0x0 | DQS gating calibration CS select signal 2'b00: select CS0 and CS1 2'b01: select CS1 2'b10: select CS0 |
| | 1 | 0x0 | set calibration bypass mode(1:bypass mode; 0:nomal) |
| | 0 | 0x0 | set calibration start (1: start; 0: stop) |

4. Wait for the calibration finish by PHYREGFF.
5. Normal read and writes operation can begin.

2.6.5 DDR PHY Write Leveling Training

DDR PHY auto write leveling training function has been implemented in the PHY. The entire training processes only need to configure the register to start and wait for finish.

The entire training process is as follows:

1. PHY's register is reset, the setup is complete.
2. Send the initial command to dram and complete dram initialization.
3. Set the PHYREG05 and PHYREG06 to configure the dram mode register which used to enable dram write leveling training function.
4. Set the PHY's register to begin training.

| Offset | Bit | Default | Description |
|--------|-----|---------|--|
| 0x8 | 7~6 | 0x0 | Write leveling CS select signal 2'b00: select CS0 and CS1 2'b01: select CS1 2'b10: select CS0 |
| | 3 | 0x0 | Write leveling calibration bypass mode, active high |
| | 2 | 0x0 | Write leveling calibration control, active high |

5. Wait for the calibration finish by PHYREGF0.
6. Normal read and writes operation can begin.

2.6.6 DDR Standby Mode

The standby mode is enabled by register DDR_STDBY_CONTROL0 [0]. When DDR controller is idle, and after a period of waiting time, the standby mode will be activated, the clocks of DDR controller, PHY and memory scheduler can be gated. The waiting time is determined by the register DDR_STDBY_CONTROL0 [31:16]. It is the counter threshold by controller clock.

The register DDR_STDBY_CONTROL0[7:4] is used to determine if the clock of memory schedule, PHYCTL or DDR controller will be gated when in standby status.

2.6.7 DDR Monitor

1. DDR read or write address monitor

DDR monitor module can store 4 consecutive read or write addresses in real time. We can read these addresses by APB bus for debug when system enters abnormal state.

The steps of configuration to monitor DDR read or write address:

- Configure DDRMON_DDR_IF_CTRL.direction to select storing read or write address.
- Set DDRMON_DDR_IF_CTRL.if_mon_en to '1' to enable DDR monitor.
- When system is abnormal, we can read the register to get the current four addresses.

2. DDR access address monitor within a specified range

Sometimes we want to confirm whether DDR read or write within a specified address range, then we can configure the address range and enable this function.

The steps of configuration to monitor DDR access address within a specified range:

Configure the write address range registers DDRMON_W_R_START_ADDR, DDRMON_W_R_END_ADDR, and read address registers DDRMON_R_D_START_ADDR, DDRMON_R_D_END_ADDR.

- Enable interrupt by configure the register DDRMON_INT_MASK[6:5] to 0.
- Set DDRMON_DDR_IF_CTRL.if_mon_en to '1' to enable DDR address monitor.
- If the read or write addresses hit the range, then interrupt will assert, and we can read the interrupt status register DDRMON_INT_STATUS.

3. DDR access command statistics

This module can do the statistics about DDR access command, like write, read and active by monitoring DFI interface. There are two mode to do statistics, hardware mode and software mode. Two thresholds can be set, if read and write command number is more than high threshold, or less than low threshold, the interrupt will be asserted.

4. DDR low power statistics

It does the statistics the low power period such as DDR self-refresh, power down, clkstop and PHY low power. After compare statistics interrupt asserts, the low power period represent in correspond count register will be updated and can be accessed through APB bus interface.

Hardware mode

In hardware mode, a dfi timer is used to specify a statistics period, the command statistics is done in the statistics period. The dfi timer is running in 24MHz. After dfi timer counts to the threshold, and update the statistics value, the dfi timer will restart automatically, and count again.

The steps of hardware mode of DDR access command statistics:

- Configure register DDRMON_CTRL.hardware_en as '1' to enable hardware mode.
- Configure register DDRMON_TIMER_COUNT to set the dfi timer count threshold, the statistics is done in the period of timer being less than the value of DDRMON_TIMER_COUNT.
- Configure register DDRMON_CTRL.lpddr23_en and DDRMON_CTRL.lpddr4_en to set the DDR mode:

| DDRMON_CTRL.lpddr23_en | DDRMON_CTRL.lpddr4_en | DDR mode |
|-------------------------------|------------------------------|-----------------|
| 1 | 0 | LPDDR2/LPDDR3 |
| 0 | 0 | DDR3 |
| 0 | 1 | LPDDR4 |

- Configure register DDRMON_FLOOR_NUMBER to specify the low threshold of interrupt, and configure register DDRMON_TOP_NUMBER to specify the high threshold of interrupt.
- Configure register DDRMON_CTRL.timer_cnt_en as '1' to start hardware mode.
- Wait for the interrupt to do following process. We also can read the read, write and active command number separately.

Software mode

In software mode, the statistics is controlled by software.

The steps of hardware mode of DDR access command statistics:

- Configure register DDRMON_CTRL.lpddr23_en and DDRMON_CTRL.lpddr4_en to set the DDR mode like hardware mode.
- Configure register DDRMON_CTRL.software_en as '1' to enable software mode statistics.
- Configure register DDRMON_CTRL.software_en as '0' to stop the statistics, and generate the statistics result. We can read the read, write and active command number separately.

Chapter 3 Encryption and Decryption(Crypto)

3.1 Overview

Crypto is a hardware accelerator for encrypting or decrypting. It supports the most commonly used algorithm: AES, SHA1, SHA256, MD5 and PKA.

The Crypto supports following features:

- Support Link List Item (LLI) DMA transfer
- Support SHA-1, SHA-256/224, MD5 with hardware padding
- Support HMAC of SHA-1, SHA-256, MD5 with hardware padding
- Support AES-128 encrypt & decrypt cipher
- Support AES ECB/CBC/OFB/CFB/CTR/CTS/XTS mode
- Support up to 4096 bits PKA mathematical operations for RSA/ECC
- Support up to 8-channels configuration
- Support Up to 256 bits TRNG Output

3.2 Block Diagram

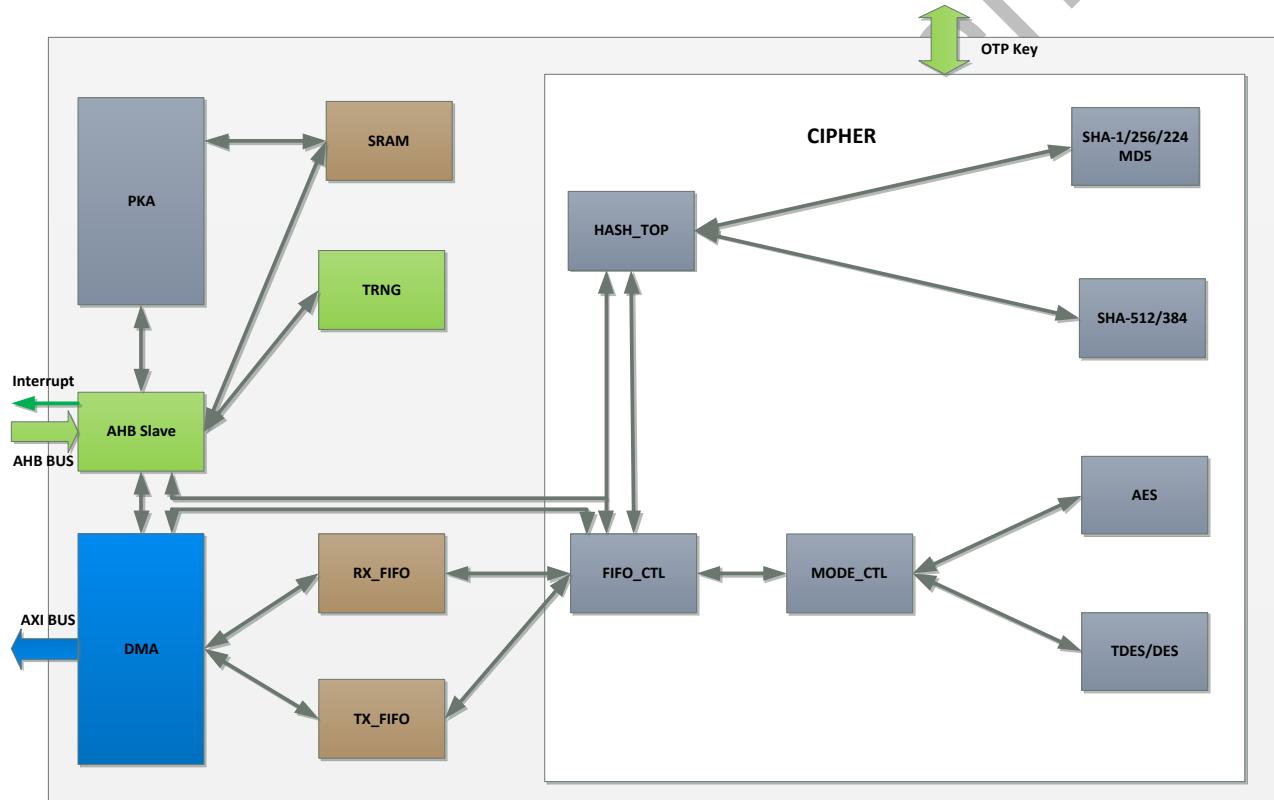


Fig. 3-1 Crypto Architecture

Crypto contains several modules : AHB_Slave, DMA, CIPHER, PKA, TRNG.

AHB_Slave

AHB_Slave is used to configure registers. This module is in HCLK domain.

DMA

DMA is used to transfer data from external memory to RX_FIFO, or from TX_FIFO to external memory. DMA uses 64-bits AXI3 protocol with max burst length to 16. LLI transfer is also supported for performance and convenience consideration. This module is in ACLK domain.

CIPHER

CIPHER contains AES, DES/TDES and HASH engines. And it also supports various mode operations. The source data is either from RX_FIFO , or from other engine output. The result data is sending either to TX_FIFO, or Registers in module AHB_Slave. This module is in CLK_CORE domain.

PKA

PKA is used to accelerate mathematical operations for big numbers. It supports - Modular arithmetic (addition, subtraction, multiplication and division), Regular arithmetic (addition, subtraction, multiplication and division), Modular inversion, Modular exponentiation, Logical operations (AND, OR, XOR, SHIFT). PKA has a SRAM which is used to store source , result and intermediate data for PKA operations. The software driver could use PKA operations to implement complicate calculation, such as RSA, ECC etc. It could support up to 4096 bits RSA modular exponentiation calculation. This module is in CLK_PKA domain.

TRNG

TRNG is used to collects random bits from the ring oscillator, up to 256 random bits per time. This module is in HCLK domain.

3.3 Register description

3.3.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

3.3.2 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|--------------------------|--------|------|-------------|--|
| CRYPTO_CLK_CTL | 0x0000 | W | 0x00000001 | Clock Control Register |
| CRYPTO_RST_CTL | 0x0004 | W | 0x00000000 | Reset Control Register |
| CRYPTO_DMA_INT_EN | 0x0008 | W | 0x00000000 | DMA Interrupt Enable Register |
| CRYPTO_DMA_INT_ST | 0x000c | W | 0x00000000 | DMA Interrupt Status Register |
| CRYPTO_DMA_CTL | 0x0010 | W | 0x00000000 | DMA Control Register |
| CRYPTO_DMA_LLI_ADDR | 0x0014 | W | 0x00000000 | DMA LIST Start Address Register |
| CRYPTO_DMA_ST | 0x0018 | W | 0x00000000 | DMA Status Register |
| CRYPTO_DMA_STATE | 0x001c | W | 0x00000000 | DMA State Register |
| CRYPTO_DMA_LLI_RADDR | 0x0020 | W | 0x00000000 | DMA LLI Read Address Register |
| CRYPTO_DMA_SRC_RADD R | 0x0024 | W | 0x00000000 | DMA Source Data Read Address Register |
| CRYPTO_DMA_DST_WAD DR | 0x0028 | W | 0x00000000 | DMA Destination Data Read Address Register |
| CRYPTO_DMA_ITEM_ID | 0x002c | W | 0x00000000 | DMA Descriptor ID Register |
| CRYPTO_FIFO_CTL | 0x0040 | W | 0x00000003 | FIFO Control Register |
| CRYPTO_BC_CTL | 0x0044 | W | 0x00000000 | Block Cipher Control Register |
| CRYPTO_HASH_CTL | 0x0048 | W | 0x00000004 | Hash Control Register |
| CRYPTO_CIPHER_ST | 0x004c | W | 0x00000000 | Cipher Status Register |
| CRYPTO_CIPHER_STATE | 0x0050 | W | 0x00000400 | Cipher Current State Register |
| CRYPTO_CHn_IV_0 | 0x0100 | W | 0x00000000 | Channel n IV Register 0 |
| CRYPTO_CHn_IV_1 | 0x0104 | W | 0x00000000 | Channel n IV Register 1 |
| CRYPTO_CHn_IV_2 | 0x0108 | W | 0x00000000 | Channel n IV Register 2 |
| CRYPTO_CHn_IV_3 | 0x010c | W | 0x00000000 | Channel n IV Register 3 |
| CRYPTO_CHn_KEY_0 | 0x0180 | W | 0x00000000 | Channel n KEY Register 0 |
| CRYPTO_CHn_KEY_1 | 0x0184 | W | 0x00000000 | Channel n KEY Register 1 |
| CRYPTO_CHn_KEY_2 | 0x0188 | W | 0x00000000 | Channel n KEY Register 2 |
| CRYPTO_CHn_KEY_3 | 0x018c | W | 0x00000000 | Channel n KEY Register 3 |
| CRYPTO_HASH_DOUT_0 | 0x03a0 | W | 0x00000000 | HASH Data Output Register 0 |

| Name | Offset | Size | Reset Value | Description |
|-----------------------|--------|------|-------------|--------------------------------|
| CRYPTO HASH DOUT_1 | 0x03a4 | W | 0x00000000 | HASH Data Output Register 1 |
| CRYPTO HASH DOUT_2 | 0x03a8 | W | 0x00000000 | HASH Data Output Register 2 |
| CRYPTO HASH DOUT_3 | 0x03ac | W | 0x00000000 | HASH Data Output Register 3 |
| CRYPTO HASH DOUT_4 | 0x03b0 | W | 0x00000000 | HASH Data Output Register 4 |
| CRYPTO HASH DOUT_5 | 0x03b4 | W | 0x00000000 | HASH Data Output Register 5 |
| CRYPTO HASH DOUT_6 | 0x03b8 | W | 0x00000000 | HASH Data Output Register 6 |
| CRYPTO HASH DOUT_7 | 0x03bc | W | 0x00000000 | HASH Data Output Register 7 |
| CRYPTO HASH VALID | 0x03e4 | W | 0x00000000 | HASH Output Valid Register |
| CRYPTO VERSION | 0x03f0 | W | 0x00000000 | CRYPTO Version Number Register |
| CRYPTO RNG CTL | 0x0400 | W | 0x0000000c | RNG Control Register |
| CRYPTO RNG SAMPLE_CNT | 0x0404 | W | 0x00000000 | RNG Sample Counter Register |
| CRYPTO RNG DOUT_0 | 0x0410 | W | 0x00000000 | RNG Data Output Register 0 |
| CRYPTO RNG DOUT_1 | 0x0414 | W | 0x00000000 | RNG Data Output Register 1 |
| CRYPTO RNG DOUT_2 | 0x0418 | W | 0x00000000 | RNG Data Output Register 2 |
| CRYPTO RNG DOUT_3 | 0x041c | W | 0x00000000 | RNG Data Output Register 3 |
| CRYPTO RNG DOUT_4 | 0x0420 | W | 0x00000000 | RNG Data Output Register 4 |
| CRYPTO RNG DOUT_5 | 0x0424 | W | 0x00000000 | RNG Data Output Register 5 |
| CRYPTO RNG DOUT_6 | 0x0428 | W | 0x00000000 | RNG Data Output Register 6 |
| CRYPTO RNG DOUT_7 | 0x042c | W | 0x00000000 | RNG Data Output Register 7 |
| CRYPTO RAM CTL | 0x0480 | W | 0x00000000 | RAM Control Register |
| CRYPTO RAM ST | 0x0484 | W | 0x00000001 | RAM Status Register |
| CRYPTO DEBUG CTL | 0x04a0 | W | 0x00000000 | PKA Debug Control Register |
| CRYPTO DEBUG ST | 0x04a4 | W | 0x00000001 | PKA Debug Status Register |
| CRYPTO DEBUG MONITOR | 0x04a8 | W | 0x0000feef | PKA Debug Monitor Bus Register |
| CRYPTO PKA MEM MAP0 | 0x0800 | W | 0x00000000 | PKA Memory Map 0 Register |
| CRYPTO PKA MEM MAP1 | 0x0804 | W | 0x00000000 | PKA Memory Map 1 Register |
| CRYPTO PKA MEM MAP2 | 0x0808 | W | 0x00000000 | PKA Memory Map 2 Register |
| CRYPTO PKA MEM MAP3 | 0x080c | W | 0x00000000 | PKA Memory Map 3 Register |
| CRYPTO PKA MEM MAP4 | 0x0810 | W | 0x00000000 | PKA Memory Map 4 Register |
| CRYPTO PKA MEM MAP5 | 0x0814 | W | 0x00000000 | PKA Memory Map 5 Register |
| CRYPTO PKA MEM MAP6 | 0x0818 | W | 0x00000000 | PKA Memory Map 6 Register |
| CRYPTO PKA MEM MAP7 | 0x081c | W | 0x00000000 | PKA Memory Map 7 Register |
| CRYPTO PKA MEM MAP8 | 0x0820 | W | 0x00000000 | PKA Memory Map 8 Register |
| CRYPTO PKA MEM MAP9 | 0x0824 | W | 0x00000000 | PKA Memory Map 9 Register |
| CRYPTO PKA MEM MAP10 | 0x0828 | W | 0x00000000 | PKA Memory Map 10 Register |
| CRYPTO PKA MEM MAP11 | 0x082c | W | 0x00000000 | PKA Memory Map 11 Register |
| CRYPTO PKA MEM MAP12 | 0x0830 | W | 0x00000000 | PKA Memory Map 12 Register |

| Name | Offset | Size | Reset Value | Description |
|-------------------------|--------|------|-------------|-----------------------------|
| CRYPTO PKA MEM MAP1_3 | 0x0834 | W | 0x00000000 | PKA Memory Map 13 Register |
| CRYPTO PKA MEM MAP1_4 | 0x0838 | W | 0x00000000 | PKA Memory Map 14 Register |
| CRYPTO PKA MEM MAP1_5 | 0x083c | W | 0x00000000 | PKA Memory Map 15 Register |
| CRYPTO PKA MEM MAP1_6 | 0x0840 | W | 0x00000000 | PKA Memory Map 16 Register |
| CRYPTO PKA MEM MAP1_7 | 0x0844 | W | 0x00000000 | PKA Memory Map 17 Register |
| CRYPTO PKA MEM MAP1_8 | 0x0848 | W | 0x00000000 | PKA Memory Map 18 Register |
| CRYPTO PKA MEM MAP1_9 | 0x084c | W | 0x00000000 | PKA Memory Map 19 Register |
| CRYPTO PKA MEM MAP2_0 | 0x0850 | W | 0x00000000 | PKA Memory Map 20 Register |
| CRYPTO PKA MEM MAP2_1 | 0x0854 | W | 0x00000000 | PKA Memory Map 21 Register |
| CRYPTO PKA MEM MAP2_2 | 0x0858 | W | 0x00000000 | PKA Memory Map 22 Register |
| CRYPTO PKA MEM MAP2_3 | 0x085c | W | 0x00000000 | PKA Memory Map 23 Register |
| CRYPTO PKA MEM MAP2_4 | 0x0860 | W | 0x00000000 | PKA Memory Map 24 Register |
| CRYPTO PKA MEM MAP2_5 | 0x0864 | W | 0x00000000 | PKA Memory Map 25 Register |
| CRYPTO PKA MEM MAP2_6 | 0x0868 | W | 0x00000000 | PKA Memory Map 26 Register |
| CRYPTO PKA MEM MAP2_7 | 0x086c | W | 0x00000000 | PKA Memory Map 27 Register |
| CRYPTO PKA MEM MAP2_8 | 0x0870 | W | 0x00000000 | PKA Memory Map 28 Register |
| CRYPTO PKA MEM MAP2_9 | 0x0874 | W | 0x00000000 | PKA Memory Map 29 Register |
| CRYPTO PKA MEM MAP3_0 | 0x0878 | W | 0x00000000 | PKA Memory Map 30 Register |
| CRYPTO PKA MEM MAP3_1 | 0x087c | W | 0x00000000 | PKA Memory Map 31 Register |
| CRYPTO PKA OPCODE | 0x0880 | W | 0x00000000 | PKA Operation Code Register |
| CRYPTO_N_NP_TO_T1_A_DDR | 0x0884 | W | 0x000ff820 | N_NP_TO_T1_ADDR Register |
| CRYPTO PKA STATUS | 0x0888 | W | 0x00000001 | PKA Status Register |
| CRYPTO PKA SW RESET | 0x088c | W | 0x00000000 | software reset of PKA |

| Name | Offset | Size | Reset Value | Description |
|-----------------------------|--------|------|-------------|----------------------------------|
| CRYPTO PKA_L0 | 0x0890 | W | 0x00000000 | PKA Length 0 Register |
| CRYPTO PKA_L1 | 0x0894 | W | 0x00000000 | PKA Length 1 Register |
| CRYPTO PKA_L2 | 0x0898 | W | 0x00000000 | PKA Length 2 Register |
| CRYPTO PKA_L3 | 0x089c | W | 0x00000000 | PKA Length 3 Register |
| CRYPTO PKA_L4 | 0x08a0 | W | 0x00000000 | PKA Length 4 Register |
| CRYPTO PKA_L5 | 0x08a4 | W | 0x00000000 | PKA Length 5 Register |
| CRYPTO PKA_L6 | 0x08a8 | W | 0x00000000 | PKA Length 6 Register |
| CRYPTO PKA_L7 | 0x08ac | W | 0x00000000 | PKA Length 7 Register |
| CRYPTO PKA PIPE RDY | 0x08b0 | W | 0x00000001 | PKA pipe is ready for new opcode |
| CRYPTO PKA DONE | 0x08b4 | W | 0x00000001 | PKA Done Register |
| CRYPTO PKA MON SELE CT | 0x08b8 | W | 0x00000000 | PKA Monitor Select Register |
| CRYPTO PKA DEBUG RE G_EN | 0x08bc | W | 0x00000000 | PKA Debug Enable Register |
| CRYPTO DEBUG CNT AD DR | 0x08c0 | W | 0x00000000 | Debug Counter Address Register |
| CRYPTO DEBUG EXT AD DR | 0x08c4 | W | 0x00000000 | Debug Extra Address Register |
| CRYPTO PKA DEBUG HA LT | 0x08c8 | W | 0x00000000 | PKA Debug Halt State Register |
| CRYPTO PKA MON READ | 0x08d0 | W | 0x0000feef | PKA Monitor Read Register |
| CRYPTO PKA INT ENA | 0x08d4 | W | 0x00000000 | PKA Interrupt Enable Register |
| CRYPTO PKA INT ST | 0x08d8 | W | 0x00000000 | PKA Interrupt Status Register |
| CRYPTO SRAM ADDR | 0x1000 | W | 0x00000000 | SRAM Base Address |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

3.3.3 Detail Register Description

CRYPTO CLK CTL

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16 | WO | 0x0 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | auto_clkgate_en 0: disable. Symmetric Cipher and HASH Module clock will be always available; 1: enable. CRYPTO will gate unused Block Cipher and HASH module automatically to save power. |

CRYPTO_RST CTL

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:19 | RO | 0x0 | reserved |
| 18:16 | RW | 0x0 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | sw_pka_reset Software set this bit to start a reset to PKA module. After the reset is done, CRYPTO will clear this bit. |
| 1 | RW | 0x0 | sw_rng_reset Software set this bit to start a reset to TRNG module. After the reset is done, CRYPTO will clear this bit. |
| 0 | R/W SC | 0x0 | sw_cc_reset Software set this bit to start a reset to Symmetric Cipher and HASH module. After the reset is done, CRYPTO will clear this bit. |

CRYPTO DMA INT EN

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | zero_len_int_en 0: disable; 1: enable . |
| 5 | RW | 0x0 | list_err_int_en 0: disable; 1: enable . |
| 4 | RW | 0x0 | src_err_int_en 0: disable; 1: enable . |
| 3 | RW | 0x0 | dst_err_int_en 0: disable; 1: enable . |
| 2 | RW | 0x0 | src_item_done_int_en 0: disable; 1: enable . |
| 1 | RW | 0x0 | dst_item_done_int_en 0: disable; 1: enable . |
| 0 | RW | 0x0 | list_done_int_en 0: disable; 1: enable . |

CRYPTO DMA INT ST

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | reserved |
| 6 | W1 C | 0x0 | zero_len Indicate that DMA has met an 0 byte source transfer length in list descriptors. After the bit is read, the application should write 1 to clear this bit for next time use. |
| 5 | W1 C | 0x0 | list_err Indicate that DMA has met an error response when transfer list descriptors. The state machine will exit current transfer and then return to IDLE state. After the bit is read, the application should write 1 to clear this bit for next time use. |
| 4 | W1 C | 0x0 | src_err Indicate that DMA has met an error response when transfer source data. The state machine will exit current transfer and then return to IDLE state. After the bit is read, the application should write 1 to clear this bit for next time use. |
| 3 | W1 C | 0x0 | dst_err Indicate that DMA has met an error response when transfer destination data. The state machine will exit current transfer and then return to IDLE state. After the bit is read, the application should write 1 to clear this bit for next time use. |
| 2 | W1 C | 0x0 | src_item_done Indicate that DMA has completed a read transfers which the current list descriptor pointed to . After the bit is read, the application should write 1 to clear this bit for next time use. |
| 1 | W1 C | 0x0 | dst_item_done Indicate that DMA has completed a write transfers which the current list descriptor pointed to . After the bit is read, the application should write 1 to clear this bit for next time use. |
| 0 | W1 C | 0x0 | list_done Indicate that DMA has completed all the transfers which the list descriptors pointed to . After the bit is read, the application should write 1 to clear this bit for next time use. |

CRYPTO DMA CTL

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:18 | RO | 0x0 | reserved |
| 17:16 | WO | 0x0 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:2 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 1 | R/W SC | 0x0 | dma_restart If DMA data for next stage is not ready, application could pause DMA by descriptor commands. DMA will stop prefetching next descriptor . The application could restart DMA by asserting this bit when DMA data for next state is ready. Crypto will continue with previous transfer, and clear the bit automatically. |
| 0 | R/W SC | 0x0 | dma_start DMA asserts the bit to start DMA transfer, then Crypto will clear the bit automatically . |

CRYPTO DMA LLI ADDR

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:3 | RW | 0x00000000 | dma_lli_addr When DMA_CTL.start asserted, Crypto will read the address to get the 1'st descriptor. It should be 8-bytes align. We suggest dma_lli_addr 64-byte align for best performance consideration. |
| 2:0 | RO | 0x0 | reserved |

CRYPTO DMA ST

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | dma_busy 0: dma idle ; 1: dma busy . |

CRYPTO DMA STATE

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | reserved |
| 5:4 | RO | 0x0 | dma_lli_state For debug use only. 00: IDLE STATE; 01: FETCH STATE ; 10: WORK STATE ; |
| 3:2 | RO | 0x0 | dma_src_state For debug use only. 00: IDLE STATE; 01: LOAD STATE ; 10: WORK STATE ; |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 1:0 | RO | 0x0 | dma_dst_state For debug use only. 00: IDLE STATE; 01: LOAD STATE ; 10: WORK STATE ; |

CRYPTO DMA LLI RADDR

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | dma_lli_raddr For debug use only. It indicates the current dma lli read address. |

CRYPTO DMA SRC RADDR

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | dma_src_raddr For debug use only. It indicates the current dma source read address. |

CRYPTO DMA DST WADDR

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | dma_dst_waddr For debug use only. It indicates the current dma destination write address. |

CRYPTO DMA ITEM ID

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RO | 0x00 | dma_item_id For debug use only. It indicates the current descriptor ID. |

CRYPTO FIFO CTL

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:18 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17:16 | WO | 0x0 | write_enable When bit n=1, the corresponding bit n-16 can be written by software; When bit n=0, the corresponding bit n-16 can't be written by software. |
| 15:2 | RO | 0x0 | reserved |
| 1 | RW | 0x1 | dout_byteswap 0: big endian ; 1: little endian . |
| 0 | RW | 0x1 | din_byteswap 0: big endian ; 1: little endian . |

CRYPTO BC CTL

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:26 | RO | 0x0 | reserved |
| 25:16 | WO | 0x000 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:10 | RO | 0x0 | reserved |
| 9:8 | RW | 0x0 | bc_cipher_sel 00: AES ; Others: Reserved; |
| 7:4 | RW | 0x0 | mode FOR AES, 0: ECB ; 1: CBC ; 2: CTS ; 3: CTR ; 4: CFB ; 5: OFB ; 6: XTS ; Others: Reserved. |
| 3:2 | RW | 0x0 | key_size For AES , 00: 128 bit ; Others: reserved. |
| 1 | RW | 0x0 | decrypt 0: encrypt; 1: decrypt . |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 0 | RW | 0x0 | bc_enable 0: disable; 1: enable . |

CRYPTO HASH CTL

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | WO | 0x00 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:8 | RO | 0x0 | reserved |
| 7:4 | RW | 0x0 | hash_cipher_sel 0: SHA-1 ; 1: MD-5 ; 2: SHA-256 ; 3: SHA-224 ; Others: Reserved. |
| 3 | RW | 0x0 | hmac_enable Crypto supports HMAC-SHA1, HMAC-SHA256. 0: disable; 1: enable . |
| 2 | RW | 0x1 | hw_pad_enable 0: disable; 1: enable . |
| 1 | RW | 0x0 | hash_src_sel 0: from RX-FIFO ; 1: from TX-FIFO. |
| 0 | RW | 0x0 | hash_enable 0: disable; 1: enable . |

CRYPTO CIPHER ST

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RO | 0x0 | hash_busy 0: idle ; 1: busy . |
| 0 | RO | 0x0 | block_cipher_busy 0: idle ; 1: busy . |

CRYPTO CIPHER STATE

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:15 | RO | 0x0 | reserved |
| 14:10 | RW | 0x01 | hash_state For debug use only. 00001: IDLE State ; 00010: IPAD State ; 00100: TEXT State; 01000: OPAD State; 10000: OPAD EXT State . |
| 9:8 | RW | 0x0 | reserved |
| 7:6 | RW | 0x0 | reserved |
| 5:4 | RW | 0x0 | parallel_state For debug use only. 00: IDLE State; 01: PRE State; 10: BULK State. 11: Reserved |
| 3:2 | RW | 0x0 | reserved |
| 1:0 | RO | 0x0 | serial_state For debug use only. 00: IDLE State; 01: PRE State; 10: BULK State. 11: Reserved |

CRYPTO CHn IV 0

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_iv_0 Channel n range from 0 to 7. CHn_IV_0 address = 0x0100 + 0x10 * n . For CTR Mode, IV stands for counter. For XTS Mode, IV stands for tweak. |

CRYPTO CHn IV 1

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_iv_1 Channel n range from 0 to 7. CHn_IV_1 address = 0x0104 + 0x10 * n . For CTR Mode, IV stands for counter. For XTS Mode, IV stands for tweak. |

CRYPTO CHn IV 2

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_iv_2 Channel n range from 0 to 7. CHn_IV_2 address = 0x0108 + 0x10 * n . For CTR Mode, IV stands for counter. For XTS Mode, IV stands for tweak. |

CRYPTO CHn IV 3

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_iv_3 Channel n range from 0 to 7. CHn_IV_3 address = 0x010c + 0x10 * n . For CTR Mode, IV stands for counter. For XTS Mode, IV stands for tweak. |

CRYPTO CHn KEY 0

Address: Operational Base + offset (0x0180)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_key_0 Channel n range from 0 to 7. CHn_KEY_0 address = 0x0180 + 0x10 * n . |

CRYPTO CHn KEY 1

Address: Operational Base + offset (0x0184)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_key_1 Channel n range from 0 to 7. CHn_KEY_1 address = 0x0184 + 0x10 * n . |

CRYPTO CHn KEY 2

Address: Operational Base + offset (0x0188)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_key_2 Channel n range from 0 to 7. CHn_KEY_2 address = 0x0188 + 0x10 * n . |

CRYPTO CHn KEY 3

Address: Operational Base + offset (0x018c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | chn_key_3 Channel n range from 0 to 7. CHn_KEY_3 address = 0x018c + 0x10 * n . |

CRYPTO HASH DOUT 0

Address: Operational Base + offset (0x03a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | hash_dout_0 0'th output word for all hash function, in big endian. |

CRYPTO HASH DOUT 1

Address: Operational Base + offset (0x03a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | hash_dout_1 1'th output word for all hash function, in big endian. |

CRYPTO HASH DOUT 2

Address: Operational Base + offset (0x03a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | hash_dout_2 2'th output word for all hash function, in big endian. |

CRYPTO HASH DOUT 3

Address: Operational Base + offset (0x03ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | hash_dout_3 3'th output word for all hash function, in big endian.This is MD5 last output word. HASH_DOUT_4 ~ HASH_DOUT_15 is invalid data for MD5. |

CRYPTO HASH DOUT 4

Address: Operational Base + offset (0x03b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | hash_dout_4 4'th output word for all hash function, in big endian.This is SHA-1 last output word. HASH_DOUT_5 ~ HASH_DOUT_15 is invalid data for SHA-1. |

CRYPTO HASH DOUT 5

Address: Operational Base + offset (0x03b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | hash_dout_5 5'th output word for all hash function, in big endian. |

CRYPTO HASH DOUT 6

Address: Operational Base + offset (0x03b8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | hash_dout_6 6'th output word for all hash function, in big endian. |

CRYPTO HASH DOUT 7

Address: Operational Base + offset (0x03bc)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | hash_dout_7 7'th output word for all hash function, in big endian. This is SHA-256/224 last output word. HASH_DOUT_8 ~ HASH_DOUT_15 is invalid data for SHA-256/224. |

CRYPTO HASH VALID

Address: Operational Base + offset (0x03e4)

| Bit | Attr | Reset Value | Description |
|------|---------|-------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | W1 C | 0x0 | hash_valid When HASH calculation is finished, CRYPTO will set this bit. After the bit is read by application, it should be cleared by writing 1 to this bit. 0: HASH_DOUT is invalid ; 1: HASH_DOUT is valid . |

CRYPTO VERSION

Address: Operational Base + offset (0x03f0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | version_num Version number: V1.0.0.0. |

CRYPTO RNG CTL

Address: Operational Base + offset (0x0400)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:22 | RO | 0x0 | reserved |
| 21:16 | WO | 0x00 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:6 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 5:4 | RW | 0x0 | rng_len 00: 64 bit; 01: 128 bit; 10: 192 bit; 11: 256 bit; |
| 3:2 | RW | 0x3 | ring_sel There are 4 osc rings choice to decide the rng output data. 00: fastest osc ring ; 01: slower than osc ring 0 ; 10: slower than osc ring 1 ; 11: slowest osc ring ; |
| 1 | RW | 0x0 | rng_enable 0: disable; 1: enable . |
| 0 | R/W SC | 0x0 | rng_start The application triggers this bit to start collect rng output data. After rng is started, CRYPTO will clear the bit automatically. 0: do nothing; 1: start . |

CRYPTO RNG SAMPLE CNT

Address: Operational Base + offset (0x0404)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | reserved |
| 15:0 | RW | 0x0000 | rng_sample_cnt RNG collects osc ring output bit every rng_sample_cnt time. The value of rng_sample_cnt affects RNG output data rate, the value bigger, the rate slower. |

CRYPTO RNG DOUT 0

Address: Operational Base + offset (0x0410)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | rng_dout_0 The 32'th osc ring bit is captured in RNG_DOUT_0.bit31. |

CRYPTO RNG DOUT 1

Address: Operational Base + offset (0x0414)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | rng_dout_1 The 64'th osc ring bit is captured in RNG_DOUT_1.bit31. If RNG_CTL.rng_len = 0x00, the last valid bit of RNG is stored in RNG_DOUT_1.bit31, and RNG_DOUT_2 ~ RNG_DOUT_7 are invalid. |

CRYPTO RNG DOUT 2

Address: Operational Base + offset (0x0418)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | rng_dout_2 The 96'th osc ring bit is captured in RNG_DOUT_2.bit31. |

CRYPTO RNG DOUT 3

Address: Operational Base + offset (0x041c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | rng_dout_3 The 128'th osc ring bit is captured in RNG_DOUT_3.bit31. If RNG_CTL.rng_len = 0x01, the last valid bit of RNG is stored in RNG_DOUT_3.bit31, and RNG_DOUT_4 ~ RNG_DOUT_7 are invalid. |

CRYPTO RNG DOUT 4

Address: Operational Base + offset (0x0420)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RO | 0x00000000 | rng_dout_4 The 160'th osc ring bit is captured in RNG_DOUT_4.bit31. |

CRYPTO RNG DOUT 5

Address: Operational Base + offset (0x0424)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RO | 0x00000000 | rng_dout_5 The 192'th osc ring bit is captured in RNG_DOUT_5.bit31. If RNG_CTL.rng_len = 0x02, the last valid bit of RNG is stored in RNG_DOUT_5.bit31, and RNG_DOUT_6~ RNG_DOUT_7 are invalid. |

CRYPTO RNG DOUT 6

Address: Operational Base + offset (0x0428)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | rng_dout_6 The 224'th osc ring bit is captured in RNG_DOUT_6.bit31. |

CRYPTO RNG DOUT 7

Address: Operational Base + offset (0x042c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | rng_dout_7 The 256'th osc ring bit is captured in RNG_DOUT_7.bit31. If RNG_CTL.rng_len = 0x03, the last valid bit of RNG is stored in RNG_DOUT_7.bit31. |

CRYPTO RAM CTL

Address: Operational Base + offset (0x0480)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16 | WO | 0x0 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | ram_pka_rdy Indicate whether ram is controlled by PKA engine. 0: ram is controlled by CPU. 1: ram is controlled by CRYPTO PKA engine. |

CRYPTO RAM ST

Address: Operational Base + offset (0x0484)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x1 | clk_ram_rdy Indicate whether clk_ram is stable, and ready for use. 0: not stable ; 1: stable. |

CRYPTO DEBUG CTL

Address: Operational Base + offset (0x04a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16 | WO | 0x0 | write_enable When bit n=0, the corresponding bit n-16 can't be written by software; When bit n=1, the corresponding bit n-16 can be written by software. |
| 15:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | pka_debug_mode 0: PKA is in normal mode; 1: PKA is in debug mode . |

CRYPTO DEBUG ST

Address: Operational Base + offset (0x04a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x1 | pka_debug_clk_en For debug use only. |

CRYPTO DEBUG MONITOR

Address: Operational Base + offset (0x04a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x0000feef | pka_monitor_bus For debug use only. |

CRYPTO PKA MEM MAP0

Address: Operational Base + offset (0x0800)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map0 memory map 0 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP1

Address: Operational Base + offset (0x0804)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map1 memory map 1 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP2

Address: Operational Base + offset (0x0808)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map2 memory map 2 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP3

Address: Operational Base + offset (0x080c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map3 memory map 3 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP4

Address: Operational Base + offset (0x0810)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map4 memory map 4 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP5

Address: Operational Base + offset (0x0814)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map5 memory map 5 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP6

Address: Operational Base + offset (0x0818)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map6 memory map 6 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP7

Address: Operational Base + offset (0x081c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map7 memory map 7 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP8

Address: Operational Base + offset (0x0820)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map8 memory map 8 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP9

Address: Operational Base + offset (0x0824)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map9 memory map 9 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP10

Address: Operational Base + offset (0x0828)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map10 memory map 10 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP11

Address: Operational Base + offset (0x082c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map11 memory map 11 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP12

Address: Operational Base + offset (0x0830)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map12 memory map 12 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP13

Address: Operational Base + offset (0x0834)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map13 memory map 13 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP14

Address: Operational Base + offset (0x0838)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map14 memory map 14 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP15

Address: Operational Base + offset (0x083c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map15 memory map 15 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP16

Address: Operational Base + offset (0x0840)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map16 memory map 16 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP17

Address: Operational Base + offset (0x0844)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map17 memory map 17 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP18

Address: Operational Base + offset (0x0848)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map18 memory map 18 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP19

Address: Operational Base + offset (0x084c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map19 memory map 19 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP20

Address: Operational Base + offset (0x0850)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map20 memory map 20 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP21

Address: Operational Base + offset (0x0854)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map21 memory map 21 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP22

Address: Operational Base + offset (0x0858)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map22 memory map 22 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP23

Address: Operational Base + offset (0x085c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map23 memory map 23 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP24

Address: Operational Base + offset (0x0860)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map24 memory map 24 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP25

Address: Operational Base + offset (0x0864)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map25 memory map 25 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP26

Address: Operational Base + offset (0x0868)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map26 memory map 26 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP27

Address: Operational Base + offset (0x086c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map27 memory map 27 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP28

Address: Operational Base + offset (0x0870)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map28 memory map 28 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP29

Address: Operational Base + offset (0x0874)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map29 memory map 29 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP30

Address: Operational Base + offset (0x0878)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map30 memory map 30 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA MEM MAP31

Address: Operational Base + offset (0x087c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:2 | RW | 0x000 | memory_map31 memory map 30 [11:2], bit[1:0] is stuck to 0. |
| 1:0 | RO | 0x0 | reserved |

CRYPTO PKA OPCODE

Address: Operational Base + offset (0x0880)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | WO | 0x00 | opcode Defines the PKA operation 0x04 Add,Inc 0x05 Sub,Dec,Neg 0x06 ModAdd,ModInc 0x07 ModSub,ModDec,ModNeg 0x08 AND,TST0,CLR0 0x09 OR,COPY,SET0 0xa XOR,FLIP0,INVERT,COMPARE 0xc SHR0 0xd SHR1 0xe SHL0 0xf SHL1 0x10 MulLow 0x11 ModMul 0x12 ModMulN 0x13 ModExp 0x14 Division 0x15 Div 0x16 ModDiv 0x00 Terminate |
| 26:24 | WO | 0x0 | len The virtual length address 0-7. Virtual address 0 point to PKA_L0. Virtual address 1 point to PKA_L1. ... Virtual address 7 point to PKA_L7. |
| 23:18 | WO | 0x00 | reg_a Operand A virtual address 0-15. Virtual address 0 point to PKA_MEM_MAP0. Virtual address 1 point to PKA_MEM_MAP1. ... Virtual address 15 point to PKA_MEM_MAP15. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17:12 | WO | 0x00 | reg_b Operand B virtual address 0-15. Virtual address 0 point to PKA_MEM_MAP0. Virtual address 1 point to PKA_MEM_MAP1. ... Virtual address 15 point to PKA_MEM_MAP15. |
| 11:6 | WO | 0x00 | reg_r Result register virtual address 0-15. Virtual address 0 point to PKA_MEM_MAP0. Virtual address 1 point to PKA_MEM_MAP1. ... Virtual address 15 point to PKA_MEM_MAP15. |
| 5:0 | WO | 0x00 | tag Tag. |

CRYPTO_N NP TO T1 ADDR

Address: Operational Base + offset (0x0884)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x1f | reg_t1 Virtual address of temporary register number 1. Virtual address 0 point to PKA_MEM_MAP0. Virtual address 1 point to PKA_MEM_MAP1. ... Virtual address 15 point to PKA_MEM_MAP15. |
| 14:10 | RW | 0x1e | reg_t0 Virtual address of temporary register number 0. Virtual address 0 point to PKA_MEM_MAP0. Virtual address 1 point to PKA_MEM_MAP1. ... Virtual address 15 point to PKA_MEM_MAP15. |
| 9:5 | RW | 0x01 | reg_np Virtual address of register np. Virtual address 0 point to PKA_MEM_MAP0. Virtual address 1 point to PKA_MEM_MAP1. ... Virtual address 15 point to PKA_MEM_MAP15. |
| 4:0 | RW | 0x00 | reg_n Virtual address of register n. Virtual address 0 point to PKA_MEM_MAP0. Virtual address 1 point to PKA_MEM_MAP1. ... Virtual address 15 point to PKA_MEM_MAP15. |

CRYPTO PKA STATUS

Address: Operational Base + offset (0x0888)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:14 | RO | 0x00 | tag Tag of the Last Operation. |
| 13:9 | RO | 0x00 | opcode the last OPCODE. |
| 8 | RO | 0x0 | pka_cpu_busy PKA is busy memory control is by PKA. |
| 7 | RO | 0x0 | modinv_of_zero modular inverse of zero flag. |
| 6 | RO | 0x0 | alu_sign_out sign of the last operation(MSB). |
| 5 | RO | 0x0 | alu_carry Carry of the last ALU operation. |
| 4 | RO | 0x0 | div_by_zero Division by 0. |
| 3 | RO | 0x0 | alu_mod_ovflw Modular overflow flag. |
| 2 | RO | 0x0 | alu_out_zero ALU out is 0. |
| 1 | RO | 0x0 | pka_busy PKA is busy. |
| 0 | RO | 0x1 | pipe_is_busy PKA ready signal 0 : pipe full 1 : PKA ready for new command. |

CRYPTO PKA SW RESET

Address: Operational Base + offset (0x088c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | WO | 0x0 | pka_sw_reset PKA software reset the reset mechanism will take about four PKA clocks until the reset line is de-asserted. |

CRYPTO PKA L0

Address: Operational Base + offset (0x0890)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l0 PKA length 0, in bit unit. |

CRYPTO PKA L1

Address: Operational Base + offset (0x0894)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l1 PKA length 1, in bit unit. |

CRYPTO PKA L2

Address: Operational Base + offset (0x0898)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l2 PKA length 2, in bit unit. |

CRYPTO PKA L3

Address: Operational Base + offset (0x089c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l3 PKA length 3, in bit unit. |

CRYPTO PKA L4

Address: Operational Base + offset (0x08a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l4 PKA length 4, in bit unit. |

CRYPTO PKA L5

Address: Operational Base + offset (0x08a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l5 PKA length 5, in bit unit. |

CRYPTO PKA L6

Address: Operational Base + offset (0x08a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l6 PKA length 6, in bit unit. |

CRYPTO PKA L7

Address: Operational Base + offset (0x08ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | pka_l7 PKA length 7, in bit unit. |

CRYPTO PKA PIPE RDY

Address: Operational Base + offset (0x08b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x1 | pka_pipe_rdy PKA pipe is ready for new opcode |

CRYPTO PKA DONE

Address: Operational Base + offset (0x08b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x1 | pka_done PKA operation is completed and pipe is empty. |

CRYPTO PKA MON SELECT

Address: Operational Base + offset (0x08b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | pka_mon_select PKA monitor select which PKA fsm monitor is being output. |

CRYPTO PKA DEBUG REG EN

Address: Operational Base + offset (0x08bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | pka_debug_reg_en Enable all the debug mechanism when set. |

CRYPTO DEBUG CNT ADDR

Address: Operational Base + offset (0x08c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:0 | R/W SC | 0x00000 | debug_cnt_addr The clock counter initial values. clock is disabled when counter expires. Triggered when pka_debug_en is set. |

CRYPTO DEBUG EXT ADDR

Address: Operational Base + offset (0x08c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | WO | 0x0 | debug_ext_addr Disable the debug Mechanism |

CRYPTO_PKA_DEBUG_HALT

Address: Operational Base + offset (0x08c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x0 | pka_debug_halt In debug mode: PKA is in halt state. |

CRYPTO_PKA_MON_READ

Address: Operational Base + offset (0x08d0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x0000feef | pka_mon_read This is the PKA monitor bus register output. |

CRYPTO_PKA_INT_ENA

Address: Operational Base + offset (0x08d4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | pka_int_ena 0: disable pka interrupt; 1: enable pka interrupt. |

CRYPTO_PKA_INT_ST

Address: Operational Base + offset (0x08d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | W1C | 0x0 | pka_int_st Indicate that PKA operation completes . After the bit is read, the application should write 1 to clear this bit for next time use. |

CRYPTO_SRAM_ADDR

Address: Operational Base + offset (0x1000)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sram_addr Sram address starts from 0x1000 to 0x1fff. When RAM_CTL.ram_pka_rdy == 0, application could access sram. Otherwise, application can't . |

3.4 Application Note

3.4.1 Clock & Reset

There are 4 clock domains in Crypto. The clock and reset signals are described in the following table .

Table 3-1 Crypto Clock & Reset Description

| Signal | Attr | Description |
|-------------|-------|--|
| hclk | clock | AHB clock |
| aclk | clock | AXI master clock |
| clk_core | clock | Cipher work clock |
| clk_pka | clock | PKA work clock |
| hresetn | reset | Asynchronously assert, synchronously de-assert to hclk, low active |
| aresetn | reset | Asynchronously assert, synchronously de-assert to aclk, low active |
| resetn_core | reset | Asynchronously assert, synchronously de-assert to clk_core, low active |
| resetn_pka | reset | Asynchronously assert, synchronously de-assert to clk_pka, low active |

Each function need different clocks. The applications could gate the un-used clock to save power. Please see the following table for detail information.

Table 3-2 Crypto Clock & Reset Description

| Operation | HCLK | ACLK | CLK_CORE | CLK_PKA |
|-----------|------|------|----------|---------|
| AES | ON | ON | ON | OFF |
| HASH/HMAC | ON | ON | ON | OFF |
| PKA | ON | OFF | OFF | ON |
| TRNG | ON | OFF | OFF | OFF |

Even when CLK_CORE is on, Crypto is doing some cipher job. And Crypto could still be able to automatically gate most parts of un-used blocks to save more power, if CRYPTO_CLK_CTL.auto_clkgate_en is set to '1'. The default value for this bit is also '1'. Application could do a soft reset to a certain clock domain. Please refer to "Chapter CRU" for more details.

3.4.2 Performance

Cipher performance is shown in the following table.

Table 3-3 Crypto Performance Description

| Algorithm | block size (Byte) | clk_core frequency (Mhz) | cycle | serial max throughput (MBps) | parallel max throughput (MBps) |
|-------------|-------------------|--------------------------|-------|------------------------------|--------------------------------|
| AES-128 | 16 | 200 | 12 | 266 | 1066 |
| SHA-1 | 64 | 200 | 81 | 158 | NA |
| MD5 | 64 | 200 | 65 | 196 | NA |
| SHA-256/224 | 64 | 200 | 65 | 196 | NA |

There are 2 column throughput rates in the table, 1 is serial mode, the other is parallel mode. In parallel mode, there are 4 engines working at the same time. So the speed is 4 times than serial mode. Parallel mode includes ECB/CTR/XTS both encryption and decryption mode, CFB/CBC/CTS only decryption mode. Other modes are serial. HASH doesn't have parallel mode.

For PKA, the cycles for each calculation are not certain. It depends on the parameters. Take RSA-2048 for example, it takes about 28M cycles to finish a calculation. PKA can run 300 Mhz. It means it can run over 10 times per second.

3.4.3 DMA

DMA supports Link List Item (LLI) DMA transaction.

- Each item contains 8 bytes, and start address should be 8 bytes align ;
- We suggest that DATA start address is 8 bytes align ;
- Total DATA length is byte align.
- Support segmenting HASH/HMAC DATA into multi sections. We suggest that each section DATA length is a multiple of 64 bytes, except this section is the last section.

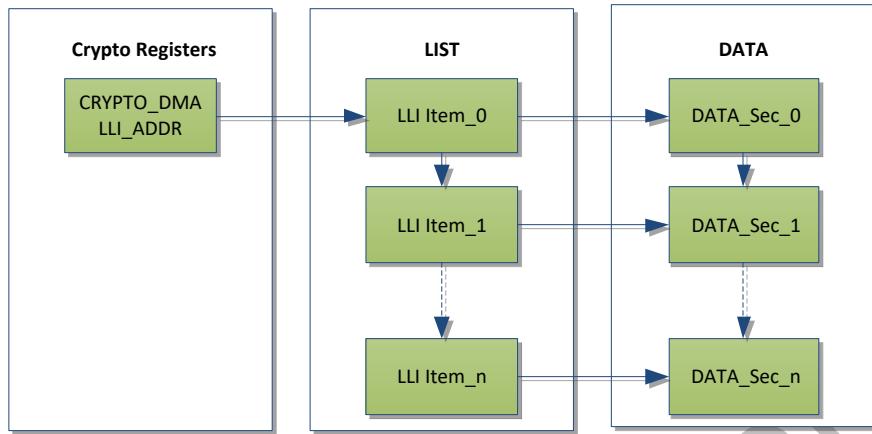


Fig. 3-2 LLI DMA Usage

As shown in the Figure above, Register CRYPTO_DMA_LLI_ADDR points to 1'st LLI item in external memory. Each LLI item contains DATA address, length, control information and next LLI item pointer, except the last LLI item. The last item doesn't have the next LLI Item pointer. After the last LLI item is finished , DMA will go to idle state.

LLI item definition is shown in the following table.

Table 3-4 LLI Item Description

| offset | Def | Description |
|--------|--------------------|---|
| 0x00 | SRC_ADDRESS[31:0] | source data start address |
| 0x04 | SRC_LENGTH[31:0] | source data length, in byte unit |
| 0x08 | DST_ADDRESS[31:0] | destination data start address |
| 0x0c | DST_LENGTH[31:0] | destination data length, in byte unit |
| 0x10 | USER_DEFINE[31:0] | used in cipher block |
| 0x14 | reserve | reserve |
| 0x18 | DMA_CTRL[31:0] | used in DMA block |
| 0x1c | next address[31:0] | next LLI item address. When DMA_CTRL.LAST = "1", NEXT_ADDRESS is invalid. |

DMA_CTRL: the definition is shown in the following table.

Table 3-5 LLI Item dma_ctl Description

| Bit | Def | Definition |
|---------|-------------------------|---|
| [31:24] | ITEM_ID[7:0] | used to identify LLI items. |
| [23:16] | reserve | |
| [15:11] | reserve | |
| 10 | source_item_done enable | When source data fetch is completed, CRYPTO_DMA_INT_ST.source_item_done will assert if this bit is set. |
| 9 | reserved | reserved |
| 8 | list_done enable | When all LLI items transfer is completed, CRYPTO_DMA_INT_ST.list_done will assert if this bit is set. |
| [7:2] | reserved | reserved |

| Bit | Def | Definition |
|-----|-------|--|
| 1 | PAUSE | indicate DMA will hold on after executes current item. DMA won't go on unless CRYPTO_DMA_CTL.restart is configured |
| 0 | LAST | indicate current item is the last one. After executes current item, DMA will return to IDLE state. |

Table 3-6 LLI Item user_define Description

| Bit | Signal | Description |
|------|--------------|--|
| 31:7 | Reserved | Reserved |
| 6:4 | Chnl_num | channel number, from 0 to 7 . |
| 3 | String_attr | indicate current item's attribution. 0: ADA ; 1: PC(plain text or cipher text) |
| 2 | String_last | indicate current item is the string last item |
| 1 | String_start | indicate current item is the string first item |
| 0 | Cipher_start | indicate current item is the cipher first item |

3.4.4 Multi-Channel Map

There are 8-channel configurations for AES or DES/TDES operation. For different key-size, the map is different. Please find the register map in the following table.

Table 3-7 LLI Item user_define Description

| Cipher sel | otpkey sel | privacy sel | chnl num | key | iv(tag/...) |
|------------|------------|-------------|----------|---------------------------|-------------------------|
| AES-128 | 0 | 0 | 0 | CH0_KEY0-3/ CH0_KEY0-1 | CH0_IV0-3/ CH0_IV0-1 |
| AES-128 | 0 | 0 | n | CHn_KEY0-3/ CHn_KEY0-1 | CHn_IV0-3/ CHn_IV0-1 |
| AES-128 | 0 | 0 | 7 | CH7_KEY0-3/ CH7_KEY0-1 | CH7_IV0-3/ CH7_IV0-1 |

In AES-XTS mode, there are 2 keys, and only AES-128 mode are. Please refer to the following table for detail information.

Table 3-8 LLI Item user_define Description

| Cipher sel | otpkey sel | privacy sel | chnl num | key1 | key2 | tweak |
|------------|------------|-------------|----------|---------------|---------------|---------------|
| AES-128 | 0 | 0 | 0 | CH0_KEY0-3 | CH4_KEY0-3 | CH0_IV0-3 |
| AES-128 | 0 | 0 | 1 | CH1_KEY0-3 | CH5_KEY0-3 | CH1_IV0-3 |
| AES-128 | 0 | 0 | 2 | CH2_KEY0-3 | CH6_KEY0-3 | CH2_IV0-3 |
| AES-128 | 0 | 0 | 3 | CH3_KEY0-3 | CH7_KEY0-3 | CH3_IV0-3 |
| AES-128 | 0 | NA | 4-7 | not supported | not supported | not supported |

3.4.5 HASH Data Path

HASH and AES could run in parallel way. There are 2 paths lead to AES-HASH function. One is AES-HASH-RX mode, the other is AES-HASH-TX mode.

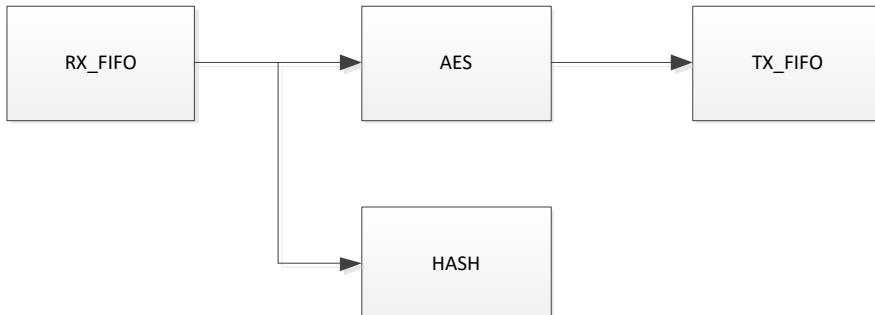


Fig. 3-3 AES-HASH-RX mode

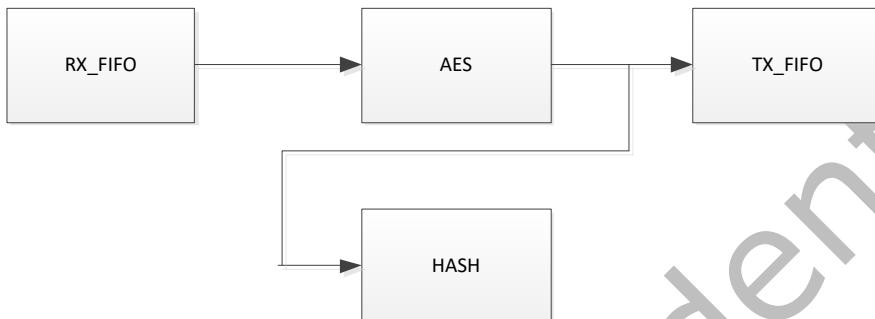


Fig. 3-4 AES-HASH-TX mode

As shown in the figures above, we could facilitate operations in some cases. For example, secure boot, we need both AES and HASH operations for the same blocks of data. The data HASH gets from RX_FIFO or TX_FIFO is byte swapped if the byte swap function is configured.

3.4.6 Program Steps

The application could succeed various crypto operations if they program properly.

- Program the LLI address to DMA_LLI_ADDR;
- Program KEY , IV, or other parameter if needed ;
- Program BC_CTL or HASH_CTL for control information;
- Prepare LLI Item;
- Enable interrupt, or do nothing.

All these operations could be in any order.

- Program DMA_CTL.start to start the operation;

This step should be the last configuration step. After this register is configured, other registers should not be changed.

- Wait interrupt asserted, or just poll the DMA_INT_ST bits.
- Program DMA_INT_ST to clear interrupt status, and get the result.

The application could also use LLI.pause when the next LLI item is not ready. After the new item is prepared, the application could program DMA_CTL.restart to continue previous operation.

Chapter 4 Multi-format Video Encoder and Decoder

4.1 Overview

The multi-format video codec (referred as codec in the following paragraph) supports h264/jpeg decoding and encoding.

The codec is connected to the AHB bus through an AHB slave and the AXI bus through an AXI master. The register configuration is fed into the decoder through the AHB slave interface while the large data such as stream data are transacted between DDR and the decoder through the AXI master interface.

In order to improve large data transaction performance, codec embeds MMU (memory management unit) and supports the cacheable bus operation.

The codec supports the following features:

- MMU embedded with MMU interrupt support
- Supports H264 decoding
 - The following profiles up to Level 4.2 : 1920x1080 @60fps
 - ◆ Baseline Profile
 - ◆ Main Profile
 - ◆ High Profile
- Supports JPEG decoding
 - 48x48 to 8176x8176(66.8 Mpixels), Step size 8 pixels
 - Baseline interleaved, and supports ROI (region of image) decode
- Supports encoding of the following standards:
 - H.264: up to HP level 4.1
 - ◆ Built-in pre-processor in common video H.264 encoder supports rotation, YCbCr conversion
 - JPEG: Baseline (DCT sequential)

4.2 Block Diagram

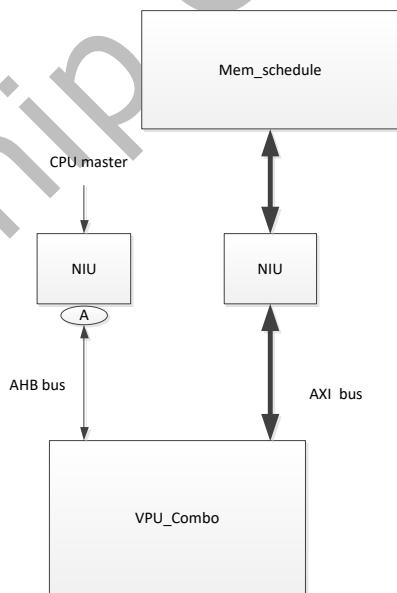


Fig. 4-1 Decoder in SOC

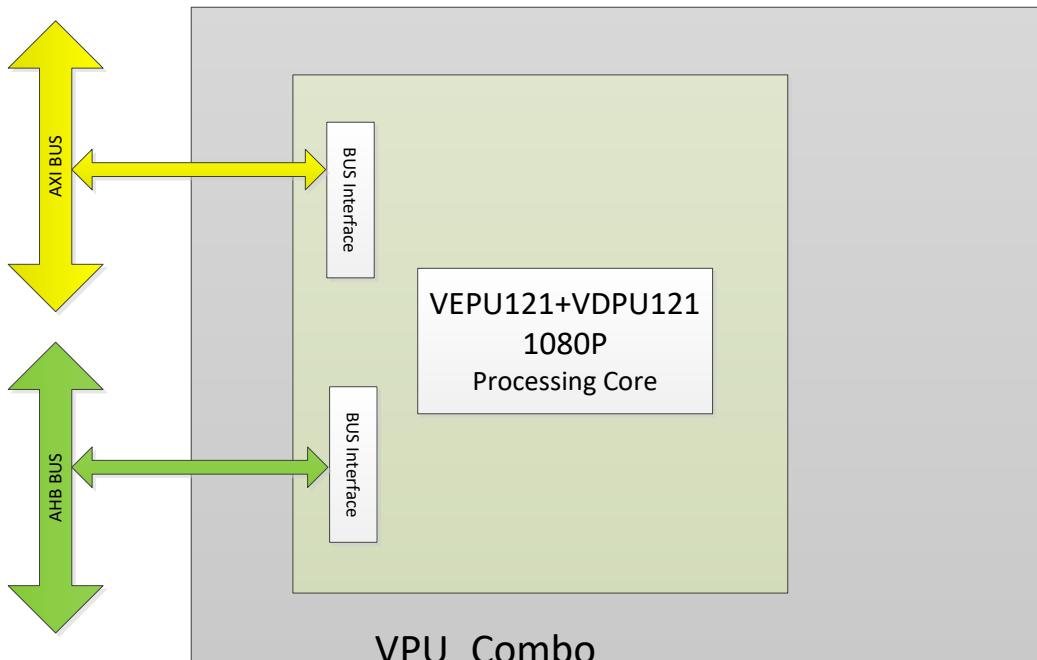


Fig. 4-2 VPU Block Diagram

As shown in the figures above, CPU accesses to the decoder register bank through AHB bus. Bitstream and other necessary data are fed into processing core through AXI read channel, and after several steps of decoding process, decoded pictures and other information data are transferred to designated location in the DDR through AXI write channel.

VDPU121 1080P processing core support multi-format decoder, such as h264 and jpeg. VEPU121 1080P processing core support h.264 and jpeg encoder.

4.3 Video frame format

This chapter describes different input and output video frame formats supported by VCODEC. Each function module has its own supported video frame formats, and this chapter describes all the video frame formats.

4.3.1 YCbCr 4:2:0 Planar Format

In the planar format, each video sample component forms one memory plane. Within one plane, the data has to be stored linearly and contiguously in the memory as shown in Fig. 4-3. The luminance samples are stored in raster-scan order (Y0Y1 Y2Y3 Y4....). The chrominance samples are stored in two planes also in raster scan order (Cb0Cb1 Cb2Cb3 Cb4.... and Cr0Cr1 Cr2Cr3 Cr4....). In this format each pixel takes 12 bits of memory.

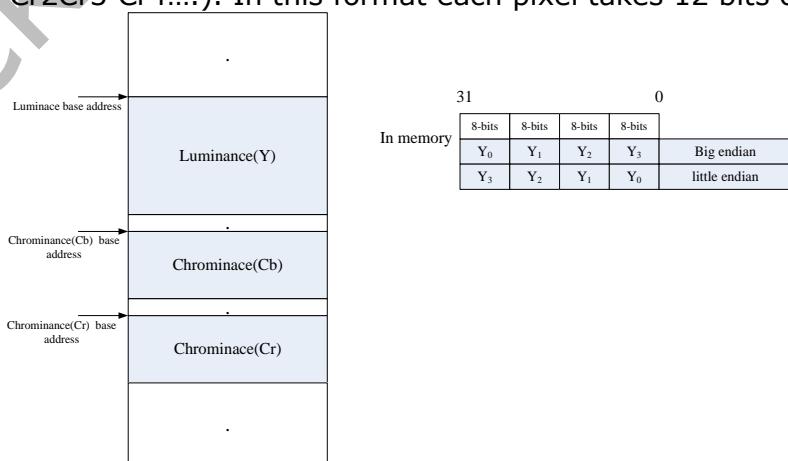


Fig. 4-3 VCODEC YCbCr 4:2:0 planar format

4.3.2 YCbCr 4:2:0 Semi-Planar format

In semi-planar YCbCr4:2:0 format the luminance samples from one plane in memory, and chrominance samples from another. Within one plane, the data has to be stored linearly and contiguously in the memory. The luminance pixels are stored in raster-scan order (Y0Y1 Y2Y3 Y4....). The interleaved chrominance CbCr samples are stored in raster-scan order in memory as Cb0Cr0 Cb1Cr1 Cb2 Cr2 Cb3Cr3 Cb4 Cr4....

Semi-Planar format supports both progressive and interlaced format as presented in Fig. 4-4. The interlaced format may be alternative line or each line.

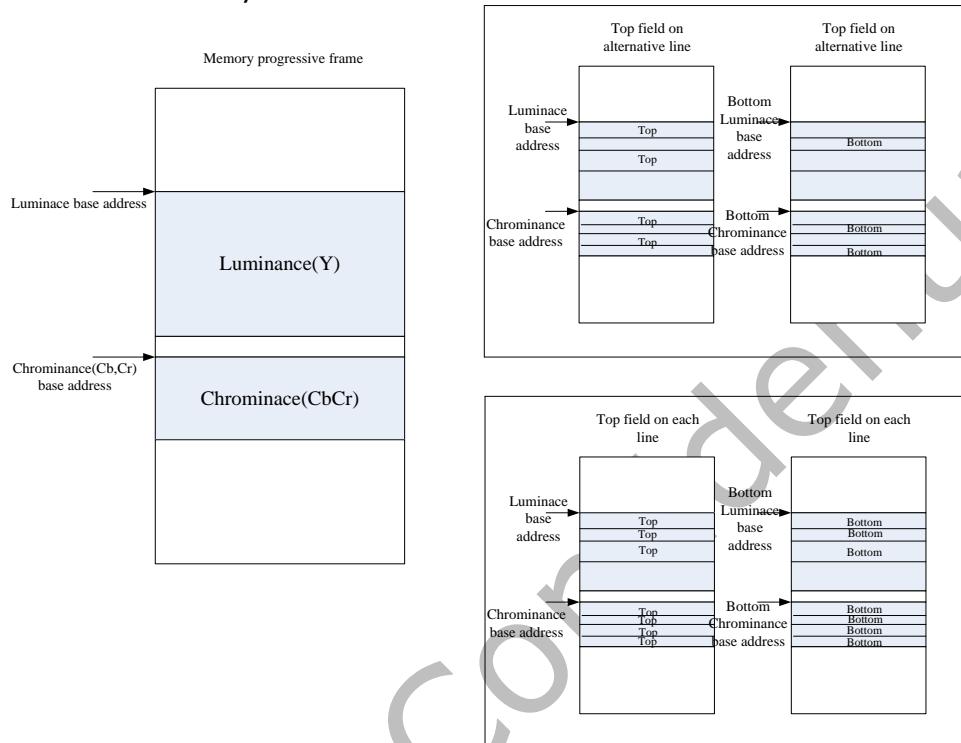


Fig. 4-4 VCODEC YCbCr 4:2:0 Semi-planar format

4.3.3 YCbCr 4:2:0 Tiled Semi-Planar Format

Like the YCbCr 4:2:0 semi-planar format, the tiled semi-planar format is also organized in the memory on two separate planes. The difference between these formats is that in tiled format the pixel samples are not anymore in raster-scan order but are stored macroblock(16x16 pixels) by macroblock. The samples of each macroblock are stored in consecutive addresses and the macroblocks are ordered from left to right and from top to down as Fig. 4-5. When this format used as input data format, it causes the lowest bus load to the system as there is minimal amount of non-sequential memory addressing required when reading the input data to the post-processor.

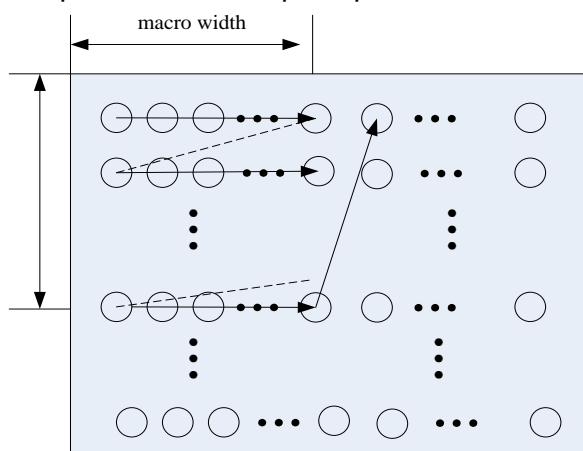


Fig. 4-5 VCODEC Tile scan mode

4.3.4 YCbCr 4:2:2 Interleaved Format

In the interleaved YCbCr 4:2:2 format the pixel samples from a single plane in which the data has to be stored linearly and contiguously as shown in Fig. 4-6. The pixel data is in raster scan order and the chrominance samples are interleaved between the luminance samples as Y0Cb0 Y1Cr0 Y2 Cb1 Y3Cr1 Y4 Cr2.... YCrCb, CbYCrY and CrYCbY component orders are supported also. In this format, each pixel takes 16 bits in the memory.

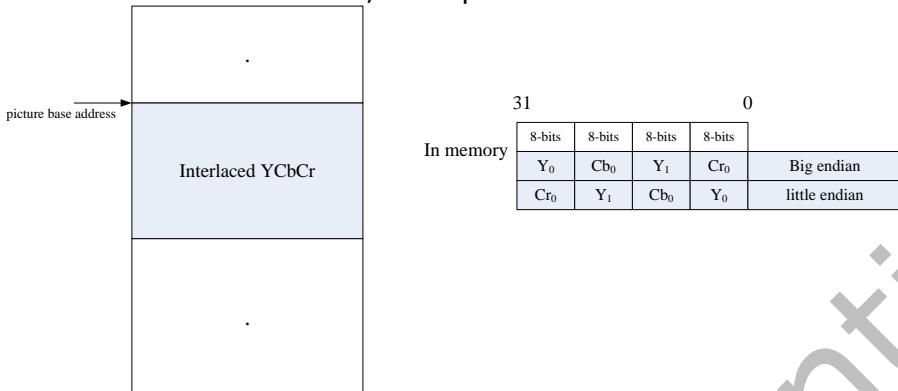


Fig. 4-6 VCODEC YCbCr4:2:2 Interleaved format

4.3.5 AYCbCr 4:4:4 Interleaved Format

In the interleaved YcbCr 4:2:2 format, the pixel samples from a single plane in which the data has to be stored linearly and contiguously as show in Fig. 4-7. The pixel data is in raster scan order and the chrominance and alpha channel samples are interleaved between the luminance samples as A0Y0 Cb0Cr0 A1 Y1 Cb1Cr1....

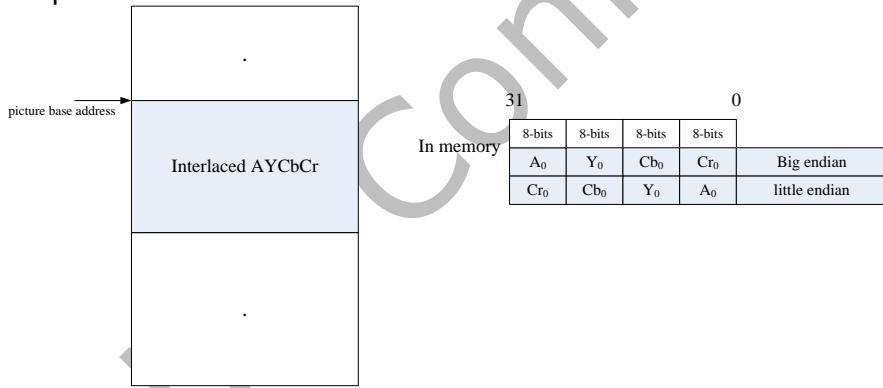


Fig. 4-7 VCODEC AYCbCr 4:4:4 Interleaved format

4.3.6 RGB 16bpp Format

In this format each pixel is represented by 16 or less bits containing the red, blue and green samples. There are several 16bpp formats which use different number of bits for each sample. For example the RGB 5-5-5 format uses 5 bits for each sample and 1 bit is left unused or can represent a transparency flag, where RGB 5-6-5 uses 6 bits for the G sample and 5 bits for R and B samples. Common for all 16bpp types is that two pixels fit into one 32-bit space.

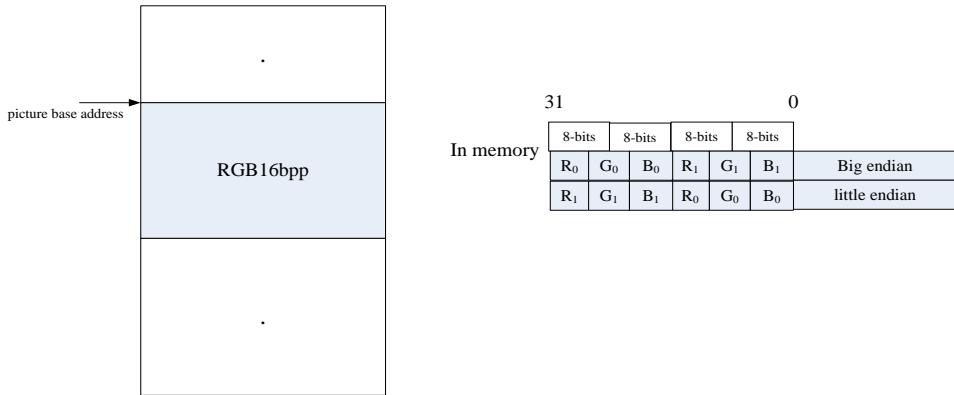


Fig. 4-8 VCODEC RGB 16bpp format

4.3.7 RGB 32bpp Format

Any RGB format that has its pixels represented by more than 16bits each is considered to be of 32bpp type. Typically in this format each pixel is represented by three bytes containing a red, blue and green sample and a 4th byte which can be empty or hold an alpha blending value. Common for all 32bpp types is that only one pixel fit into one 32-bit space. The data has to be stored linearly and contiguously in the memory.

4.4 Function Description

4.4.1 MMU

The MMU divides memory into 4KB pages, where each page can be individually configured. For each page the following parameters are specified:

- Address translation of virtual memory, this enables the processor to work using address that differ from the physical address in the memory system.
- The permitted types of accesses to that page. Each page can permit read, write, both, or none.

The MMU use 2-level page table structure:

1. The first level, the page directory consists of 1024 directory table entries(DTEs), each pointing to a page table.
2. The second level, the page table consists of 1024 page table entries(PTEs), each pointing to a page in memory.

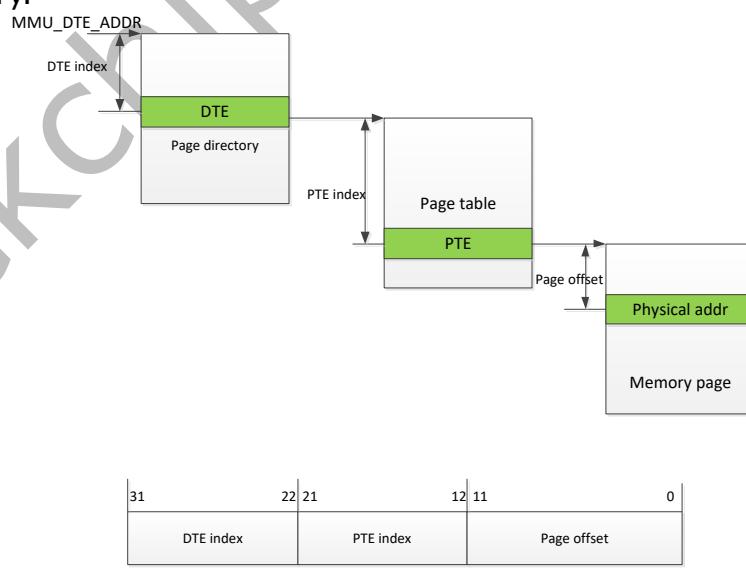


Fig. 4-9 Structure of two-level page table

4.4.2 H264 decoder

The H264 feature that Video decoder supports shows as below.

Table 4-1 H.264 feature

| Feature | Decoder support |
|---------|-----------------|
|---------|-----------------|

| | |
|---------------------------------|---|
| Input data format | H.264 byte or NAL unit stream /SVC stream /MVC stream |
| Decoding scheme | Frame by frame(or field by field) Slice by Slice |
| Output data format | YCbCr 4:2:0 semi-planar |
| Supported image size | 48x48 to 1920x1088 Step size 16 pixels |
| Maximum frame rate | 60fps at 1920x1088 |
| Maximum bit rate | As specified by H.264 HP level 4.2 |
| Error detection and concealment | Supported |

The input of the decoder is H.264 standard bit stream in either plain NAL unit format or byte stream format. The input format in use will be automatically detected. The H.264 video encoding allows the use of multiple reference pictures, which means that the decoding order of the pictures may be different from their display order. The decoder can perform internally the display reordering of the decoded pictures or it can skip this and output all the pictures as soon as they are decoded.

The decoder has two operating modes: in the primary mode the HW performs entropy decoding, and in the secondary mode SW performs entropy decoding. Secondary mode is used in H.264 ASO or Slice Group stream decoding.

4.4.3 JPEG Decoder

JPEG features supported by decoder are as shown in Table 4-2.

Table 4-2 JPEG features

| Feature | Decoder support |
|----------------------|---|
| Input data format | JFIF file format 1.02 YCbCr 4:0:0, 4:2:0, 4:2:2, 4:4:0, 4:1:1 and 4:4:4 semi-planar |
| Decoding scheme | Input: buffer by buffer, from 5Kb to 8MB at a time ^① Output: from 1 MB row to 16 Mpixels at a time ^② |
| Output data format | YCbCr 4:0:0, 4:2:0, 4:2:2, 4:4:0, 4:1:1 and 4:4:4 semi-planar |
| Supported image size | 48x48 to 8176x8176(66.8 Mpixels) Step size 8 pixels ^③ |
| Maximum frame rate | Up to 76 million pixels pre second |
| Maximum bit rate | As specified by the specification |
| Thumbnail decoding | JPEG compressed thumbnails supported |
| Error detection | Supported |

Notes:

^①Programmable buffer size for optimizing performance and memory consumption. Interrupt will be issued when buffer runs empty, and the control software will load more streams to external memory.

^②Programmable output slice for optimizing performance and memory consumption. Interrupt will be issued when the requested area decoded. The control software can be used to switch the decoder output address each time.

^③Non-16x16 dividable resolutions will be filled to 16 pixel boundary.

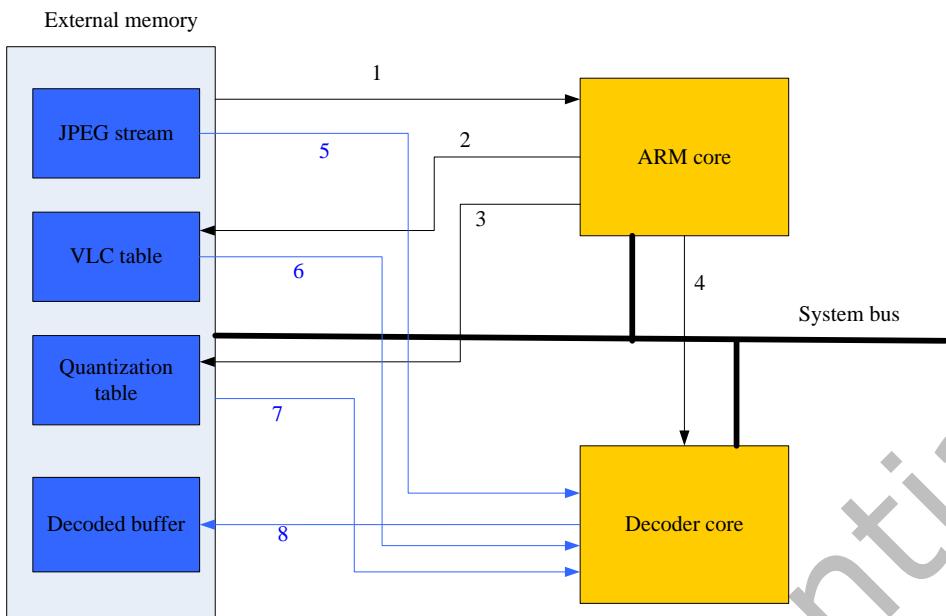


Fig. 4-10 The dataflow of JPEG decoder

The data flow of jpeg decoder is as Fig. 4-10 shown. The decoder software starts decoding the picture by parsing the stream headers (1) and then writes the following items to external memory:

VLC tables (2)

Quantization tables (3)

Last step for the software is to write the hardware control registers and to enable the hardware (4). After starting hardware, SW waits interrupt from HW.

Hardware decodes the picture by reading stream (5), VLC (6) and QP (7) tables. Hardware writes the decoded output picture memory one macroblock at a time (8). When the picture has been fully decoded, or the hardware has run out of stream data, it gives an interrupt with a proper status flag and provides stream end address for software to continue and returns to initial state.

4.4.4 H.264 Encoder

The H.264 features supported by encoder are as shown in Table 4-3 .

Table 4-3 Video encoder H.264 feature

| Feature | Encoder support |
|----------------------|--|
| Input data format | YCbCr formats: YCbCr 4:2:0 planar YCbCr 4:2:0 semi-planar YCbYCr 4:2:2① CbYCrY 4:2:2 Interleaved① RGB formats:① RGB444 to BGR444 RGB555 to BGR555 RGB565 to BGR565 RGB888 to BRG888 RGB101010 and BRG 101010 |
| Output data format | H.264: Byte unit stream NAL unit stream |
| Supported image size | 96x96 to 1920x1080(Full HD) Step size 4 pixels |

| | |
|--------------------|--|
| Maximum frame rate | 30 fps at 1920 x1080 |
| Bit rate | Maximum 20Mbps Minimum 10kbps |

①internally encoder handles image only in 4:2:0 format

Figure Fig. 4-11 illustrates the encoder data flow in H.264 encoding mode. The numbers present the following transactions:

Memory-mapped register writes and reads

Input image read

Reference image write

Reference image read

NAL sizes write from HW

NAL sizes read to SW

Output byte or NAL unit stream write from HW

Output byte or NAL unit stream headers write from SW

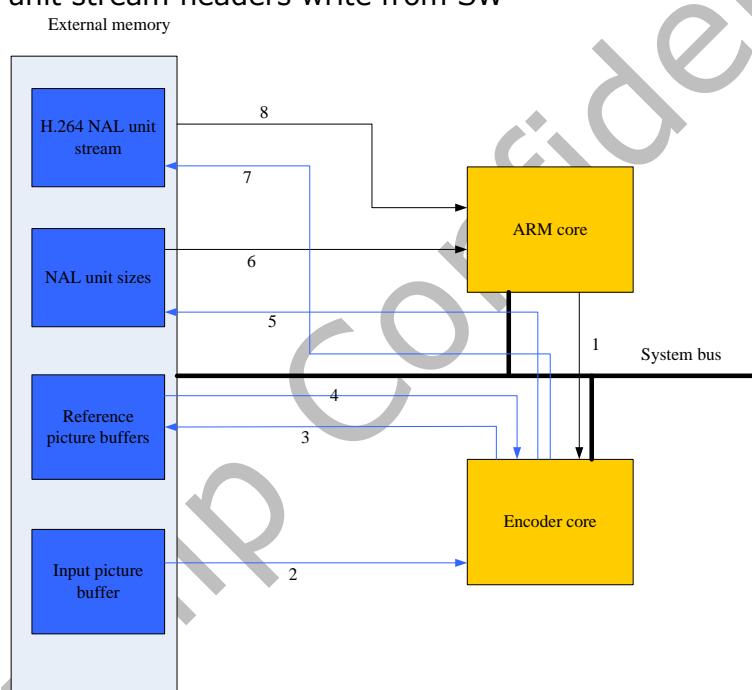


Fig. 4-11 Video Encoder Dataflow

The encoder software starts encoding the first picture by initializing hardware and writing the stream headers. After HW has encoded the image, SW calculates new quantization values for HW, and initializes HW again.

4.4.5 JPEG Encoder

The JPEG features supported by the encoder are as shown as follows:

Table 4-4 JPGE features

| Feature | Encoder support |
|-------------------|---|
| Input data format | YCbCr formats: YCbCr 4:2:0 planar YCbCr 4:2:0 semi-planar YCbCr 4:2:2① CbYCrY 4:2:2 Interleaved ① RGB formats:① RGB444 and BGR444 |

| | |
|----------------------|--|
| | RGB555 and BGR555 RGB565 and BGR565 RGB888 and BRG888 RGB101010 and BRG101010 |
| Output data format | JFIF ifle format 1.02 Non-progressive JPEG |
| Supported image size | 96x32 to 8192x8192(64 million pixels) Step size 4 pixels |
| Maximum data rate | Up to 90 million pixels per second |
| Thumbnail insertion | RGB 8-bits, RGB 24-bits and JPEG compressed thumbnails supported |

①internally encoder handles images only in 4:2:0 format

4.5 Register Description

4.5.1 Internal Address Mapping

This section describes the control/status registers of the codec.

If vdpu121 decoder is chosen to work, the register base address is vdpu121_base.

If vepu121 encoder is chosen to work, the register base address is vepu121_base.

All the register config base are listed as fellows:

Table 4-5 Base address of config

| Config Register | Base addr |
|------------------------------|--------------------|
| vdpu121 function config base | vdpu121_base |
| vdpu121 mmu config base | vdpu121_base+0x400 |
| vdpu121 cache config base | vdpu121_base+0x800 |
| vepu121 function config base | vepu121_base |
| vepu121 mmu config base | vepu121_base+0x800 |

Table 4-6 Base address value

| Base addr | value |
|--------------|-------------|
| vdpu121_base | 0xFFB8_0400 |
| vepu121_base | 0xFFB8_0000 |

4.5.2 VDPU121 Function Config Register Summary

| Name | Offset | Size | Reset Value | Description |
|-------------|--------|------|-------------|--|
| VDPU_SWREG0 | 0x0000 | W | 0x00010100 | axi control |
| VDPU_SWREG1 | 0x0004 | W | 0x00000000 | color coeff register0 |
| VDPU_SWREG2 | 0x0008 | W | 0x00000000 | color coeff register1 |
| VDPU_SWREG3 | 0x000c | W | 0x00000000 | color coeff register2 |
| VDPU_SWREG4 | 0x0010 | W | 0x00000000 | scl ctrl register0 |
| VDPU_SWREG5 | 0x0014 | W | 0x00000000 | scl ctrl register1 |
| VDPU_SWREG6 | 0x0018 | W | 0x00000000 | scl ctrl register2 |
| VDPU_SWREG7 | 0x001c | W | 0x00000000 | Amount of pixels beyond border register0 |
| VDPU_SWREG8 | 0x0020 | W | 0x00000000 | Amount of pixels beyond border register2 |
| VDPU_SWREG9 | 0x0024 | W | 0x00000000 | Rmask register |

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|--|
| VDPU_SWREG10 | 0x0028 | W | 0x00000000 | Gmask register |
| VDPU_SWREG11 | 0x002c | W | 0x00000000 | Bmask register |
| VDPU_SWREG12 | 0x0030 | W | 0x00000000 | PP input picture base address for Y bottom field |
| VDPU_SWREG13 | 0x0034 | W | 0x00000000 | PP input picture base for Ch bottom field |
| VDPU_SWREG14 | 0x0038 | W | 0x00000000 | coordinate used in macroblock crop |
| VDPU_SWREG15 | 0x003c | W | 0x00000000 | range map register |
| VDPU_SWREG16 | 0x0040 | W | 0x00000000 | total num of padded for RGB |
| VDPU_SWREG17 | 0x0044 | W | 0x00000000 | hw support informan,read only |
| VDPU_SWREG18 | 0x0048 | W | 0x00000000 | base address for reading post-processing input picture luminance (top field/frame) |
| VDPU_SWREG19 | 0x004c | W | 0x00000000 | Base address for reading post-processing input picture Cb/Ch (topfield/frame) |
| VDPU_SWREG20 | 0x0050 | W | 0x00000000 | input cr component address |
| VDPU_SWREG21 | 0x0054 | W | 0x00000000 | Base address for writing post-processed picture luminance/RGB |
| VDPU_SWREG22 | 0x0058 | W | 0x00000000 | Base address for writing post-processed picture Ch |
| VDPU_SWREG23 | 0x005c | W | 0x00000000 | Display width and PP input size extension register |
| VDPU_SWREG24 | 0x0060 | W | 0x00000000 | alpha blending base address |
| VDPU_SWREG25 | 0x0064 | W | 0x00000000 | ablen of pixels scanline |
| VDPU_SWREG26 | 0x0068 | W | 0x00000000 | x-coordinate of mask area 1 for Horizontal start pixel |
| VDPU_SWREG27 | 0x006c | W | 0x00000000 | y-coordinate of mask area 1 for Horizontal start pixel |
| VDPU_SWREG28 | 0x0070 | W | 0x00000000 | x-coordinate of mask area 2 for Horizontal start pixel |
| VDPU_SWREG29 | 0x0074 | W | 0x00000000 | y-coordinate of mask area 2 for Horizontal start pixel |
| VDPU_SWREG30 | 0x0078 | W | 0x00000000 | register for deinterlace ctrl |
| VDPU_SWREG31 | 0x007c | W | 0x00000000 | contrast adjust threshold |
| VDPU_SWREG32 | 0x0080 | W | 0x00000000 | contrast adjust offset |
| VDPU_SWREG33 | 0x0084 | W | 0xfc874780 | Synthesis configuration register post-processor (read only) |
| VDPU_SWREG34 | 0x0088 | W | 0x00000000 | PP input pic size register |
| VDPU_SWREG35 | 0x008c | W | 0x00000000 | PP output pic size register |
| VDPU_SWREG36 | 0x0090 | W | 0x00000000 | the dither mode for RGB |
| VDPU_SWREG37 | 0x0094 | W | 0x00000000 | PP input/output data format |

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|--|
| VDPU_SWREG38 | 0x0098 | W | 0x00000000 | PP input/output data format |
| VDPU_SWREG39 | 0x009c | W | 0x00000000 | the display width ctrl |
| VDPU_SWREG40 | 0x00a0 | W | 0x00000000 | Register0001 Description |
| VDPU_SWREG41 | 0x00a4 | W | 0x00000008 | enable ctrl flag |
| VDPU_SWREG50 | 0x00c8 | W | 0x00000000 | video decoder ctrl register |
| VDPU_SWREG51 | 0x00cc | W | 0x00000000 | the stream length |
| VDPU_SWREG52 | 0x00d0 | W | 0x00000000 | error concealment case related |
| VDPU_SWREG53 | 0x00d4 | W | 0x00000000 | decoder format |
| VDPU_SWREG54 | 0x00d8 | W | 0x00000000 | endian for input/output data |
| VDPU_SWREG55 | 0x00dc | W | 0x00000000 | decoder int register |
| VDPU_SWREG56 | 0x00e0 | W | 0x00200101 | axi ctrl for decoder |
| VDPU_SWREG57 | 0x00e4 | W | 0x00000010 | enable flag for decoder |
| VDPU_SWREG58 | 0x00e8 | W | 0x00000000 | soft reset registerconfig addr=4E8 |
| VDPU_SWREG59 | 0x00ec | W | 0x00000000 | H264, MPEG4, VC1 Prediction filter tap |
| VDPU_SWREG60 | 0x00f0 | W | 0x00000000 | additional chrominance address |
| VDPU_SWREG61 | 0x00f4 | W | 0x00000000 | standard dependent tables start address |
| VDPU_SWREG62 | 0x00f8 | W | 0x00000000 | Direct mode motion vector write/read start address |
| VDPU_SWREG63 | 0x00fc | W | 0x00000000 | write decoder output picture or field start address |
| VDPU_SWREG64 | 0x0100 | W | 0x00000000 | rlc or vlc mode input data start addr |
| VDPU_SWREG65 | 0x0104 | W | 0x00000000 | refbufferd related |
| VDPU_SWREG66 | 0x0108 | W | 0x67312688 | ID register |
| VDPU_SWREG67 | 0x010c | W | 0xc1520000 | Synthesis configuration register decoder 1(read only) |
| VDPU_SWREG68 | 0x0110 | W | 0x00000000 | sum of partitions(read only) |
| VDPU_SWREG69 | 0x0114 | W | 0x00000000 | sum information (read only) |
| VDPU_SWREG70 | 0x0118 | W | 0x00000000 | sum of the decoded motion vector y-components(read only) |
| VDPU_SWREG71 | 0x011c | W | 0xfb356780 | information for read only register |
| VDPU_SWREG72 | 0x0120 | W | 0x00000000 | debug0 |
| VDPU_SWREG73 | 0x0124 | W | 0x00000000 | debug registers |
| VDPU_SWREG74 | 0x0128 | W | 0x00000000 | MV address for h264 |
| VDPU_SWREG75 | 0x012c | W | 0x00000000 | H.264 Intra prediction 4x4 mode start address |
| VDPU_SWREG76 | 0x0130 | W | 0x00000000 | the number of reference pic0 |
| VDPU_SWREG77 | 0x0134 | W | 0x00000000 | the number of reference pic1 |
| VDPU_SWREG78 | 0x0138 | W | 0x00000000 | the number of reference pic2 |
| VDPU_SWREG79 | 0x013c | W | 0x00000000 | the number of reference pic3 |

| Name | Offset | Size | Reset Value | Description |
|----------------------|--------|------|-------------|---|
| <u>VDPU_SWREG80</u> | 0x0140 | W | 0x00000000 | the number of reference pic4 |
| <u>VDPU_SWREG81</u> | 0x0144 | W | 0x00000000 | the number of reference pic5 |
| <u>VDPU_SWREG82</u> | 0x0148 | W | 0x00000000 | the number of reference pic6 |
| <u>VDPU_SWREG83</u> | 0x014c | W | 0x00000000 | the number of reference pic7 |
| <u>VDPU_SWREG84</u> | 0x0150 | W | 0x00000000 | reference frame0 address for h264 |
| <u>VDPU_SWREG85</u> | 0x0154 | W | 0x00000000 | reference frame1 address for h264 |
| <u>VDPU_SWREG86</u> | 0x0158 | W | 0x00000000 | reference frame2 address for h264 |
| <u>VDPU_SWREG87</u> | 0x015c | W | 0x00000000 | reference frame3 address for h264 |
| <u>VDPU_SWREG88</u> | 0x0160 | W | 0x00000000 | reference frame4 address for h264 |
| <u>VDPU_SWREG89</u> | 0x0164 | W | 0x00000000 | reference frame5 address for h264 |
| <u>VDPU_SWREG90</u> | 0x0168 | W | 0x00000000 | reference frame6 address for h264 |
| <u>VDPU_SWREG91</u> | 0x016c | W | 0x00000000 | reference frame6 address for h264 |
| <u>VDPU_SWREG92</u> | 0x0170 | W | 0x00000000 | reference frame8 address for h264 |
| <u>VDPU_SWREG93</u> | 0x0174 | W | 0x00000000 | reference frame9 address for h264 |
| <u>VDPU_SWREG94</u> | 0x0178 | W | 0x00000000 | reference frame10 address for h264 |
| <u>VDPU_SWREG95</u> | 0x017c | W | 0x00000000 | reference frame11 address for h264 |
| <u>VDPU_SWREG96</u> | 0x0180 | W | 0x00000000 | reference frame12 address for h264 |
| <u>VDPU_SWREG97</u> | 0x0184 | W | 0x00000000 | reference frame13 address for h264 |
| <u>VDPU_SWREG98</u> | 0x0188 | W | 0x00000000 | reference frame14 address for h264 |
| <u>VDPU_SWREG99</u> | 0x018c | W | 0x00000000 | reference frame15 address for h264 |
| <u>VDPU_SWREG100</u> | 0x0190 | W | 0x00000000 | initial reference picture list related0 |
| <u>VDPU_SWREG101</u> | 0x0194 | W | 0x00000000 | initial reference picture list related1 |
| <u>VDPU_SWREG102</u> | 0x0198 | W | 0x00000000 | initial reference picture list related2 |

| Name | Offset | Size | Reset Value | Description |
|---------------|--------|------|-------------|---|
| VDPU_SWREG103 | 0x019c | W | 0x00000000 | initial reference picture list related3 |
| VDPU_SWREG104 | 0x01a0 | W | 0x00000000 | initial reference picture list related4 |
| VDPU_SWREG105 | 0x01a4 | W | 0x00000000 | initial reference picture list related5 |
| VDPU_SWREG106 | 0x01a8 | W | 0x00000000 | initial reference picture list related6 |
| VDPU_SWREG107 | 0x01ac | W | 0x00000000 | long term flag for reference pictuure index |
| VDPU_SWREG108 | 0x01b0 | W | 0x00000000 | valid flag for reference picture index |
| VDPU_SWREG109 | 0x01b4 | W | 0x00000000 | the stream start word for decoder |
| VDPU_SWREG110 | 0x01b8 | W | 0x00000000 | h264 pic mb size |
| VDPU_SWREG111 | 0x01bc | W | 0x00000000 | h264 ctrl related |
| VDPU_SWREG112 | 0x01c0 | W | 0x00000000 | current frame related |
| VDPU_SWREG113 | 0x01c4 | W | 0x00000000 | reference picture related |
| VDPU_SWREG114 | 0x01c8 | W | 0x00000000 | maximum reference |
| VDPU_SWREG115 | 0x01cc | W | 0x00000000 | enable flag |
| VDPU_SWREG120 | 0x01e0 | W | 0x00000000 | multi format reuse register0 |
| VDPU_SWREG121 | 0x01e4 | W | 0x00000000 | multi format reuse register1 |
| VDPU_SWREG122 | 0x01e8 | W | 0x00000000 | multi format reuse register2 |
| VDPU_SWREG123 | 0x01ec | W | 0x00000000 | multi format reuse register3 |
| VDPU_SWREG124 | 0x01f0 | W | 0x00000000 | multi format reuse register4 |
| VDPU_SWREG125 | 0x01f4 | W | 0x00000000 | multi format reuse register5 |
| VDPU_SWREG126 | 0x01f8 | W | 0x00000000 | multi format reuse register6 |
| VDPU_SWREG127 | 0x01fc | W | 0x00000000 | multi format reuse register7 |
| VDPU_SWREG128 | 0x0200 | W | 0x00000000 | multi format reuse register8 |
| VDPU_SWREG129 | 0x0204 | W | 0x00000000 | multi format reuse register9 |
| VDPU_SWREG130 | 0x0208 | W | 0x00000000 | multi format reuse register10 |
| VDPU_SWREG131 | 0x020c | W | 0x00000000 | multi format reuse register11 |
| VDPU_SWREG132 | 0x0210 | W | 0x00000000 | multi format reuse register12 |
| VDPU_SWREG133 | 0x0214 | W | 0x00000000 | multi format reuse register13 |
| VDPU_SWREG134 | 0x0218 | W | 0x00000000 | multi format reuse register14 |
| VDPU_SWREG135 | 0x021c | W | 0x00000000 | multi format reuse register15 |
| VDPU_SWREG136 | 0x0220 | W | 0x00000000 | multi format reuse register16 |
| VDPU_SWREG137 | 0x0224 | W | 0x00000000 | multi format reuse register17 |
| VDPU_SWREG138 | 0x0228 | W | 0x00000000 | multi format reuse register18 |
| VDPU_SWREG139 | 0x022c | W | 0x00000000 | multi format reuse register19 |
| VDPU_SWREG140 | 0x0230 | W | 0x00000000 | multi format reuse register20 |
| VDPU_SWREG141 | 0x0234 | W | 0x00000000 | multi format reuse register21 |
| VDPU_SWREG142 | 0x0238 | W | 0x00000000 | multi format reuse register22 |
| VDPU_SWREG143 | 0x023c | W | 0x00000000 | multi format reuse register23 |

| Name | Offset | Size | Reset Value | Description |
|---|--------|------|-------------|---|
| VDPU_SWREG144 | 0x0240 | W | 0x00000000 | multi format reuse register24 |
| VDPU_SWREG145 | 0x0244 | W | 0x00000000 | multi format reuse register25 |
| VDPU_SWREG146 | 0x0248 | W | 0x00000000 | multi format reuse register26 |
| VDPU_SWREG147 | 0x024c | W | 0x00000000 | multi format reuse register27 |
| VDPU_SWREG148 | 0x0250 | W | 0x00000000 | multi format reuse register28 |
| VDPU_SWREG149 | 0x0254 | W | 0x00000000 | multi format reuse register29 |
| VDPU_SWREG150 | 0x0258 | W | 0x00000000 | multi format reuse register30 |
| VDPU_SWREG151 | 0x025c | W | 0x00000000 | multi format reuse register31 |
| VDPU_SWREG152 | 0x0260 | W | 0x00000000 | multi format reuse register32 |
| VDPU_SWREG153 | 0x0264 | W | 0x00000000 | multi format reuse register33 |
| VDPU_SWREG154 | 0x0268 | W | 0x00000000 | multi format reuse register34 |
| VDPU_SWREG155 | 0x026c | W | 0x00000000 | multi format reuse register35 |
| VDPU_SWREG156 | 0x0270 | W | 0x00000000 | multi format reuse register36 |
| VDPU_SWREG157 | 0x0274 | W | 0x00000000 | multi format reuse register37 |
| VDPU_SWREG158 | 0x0278 | W | 0x00000000 | multi format reuse register38 |
| VDPU_SWREG164_PERF LATENCY_CTRL0 | 0x0290 | W | 0x00000000 | Axi performance latency module ctrl register0 |
| VDPU_SWREG165_PERF LATENCY_CTRL1 | 0x0294 | W | 0x00000000 | PERF_LATENCY_CTRL1 |
| VDPU_SWREG166_PERF RD_MAX_LATENCY_NUM0 | 0x0298 | W | 0x00000000 | Read max latency number |
| VDPU_SWREG167_PERF RD_LATENCY_SAMP_NUM | 0x029c | W | 0x00000000 | The number of bigger than configed threshold value |
| VDPU_SWREG168_PERF RD_LATENCY_ACC_SUM | 0x02a0 | W | 0x00000000 | Total sample number |
| VDPU_SWREG169_PERF RD_AXI_TOTAL_BYTE | 0x02a4 | W | 0x00000000 | perf_rd_axi_total_byte |
| VDPU_SWREG170_PERF WR_AXI_TOTAL_BYTE | 0x02a8 | W | 0x00000000 | perf_wr_axi_total_byte |
| VDPU_SWREG171_PERF WORKING_CNT | 0x02ac | W | 0x00000000 | perf_working_cnt |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

4.5.3 VDPU121 Function Config Detail Register Description

VDPU_SWREG0

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x01 | sw_axi_id_wr if you config 0,will modify as 1 by hw |
| 15:8 | RW | 0x01 | sw_axi_id_rd if you config 0,will modify as 1 by hw |
| 7:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | sw_scmd_off on-off for AXI Single Command Multiple Data 0:on 1:off |
| 4:0 | RW | 0x00 | sw_max_burst_len range : 1-16 |

VDPU_SWREG1

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29:20 | RW | 0x000 | sw_coe_2st used for red color components calculate,used together with cr pix |
| 19:10 | RW | 0x000 | sw_coe_1st_1 used for all color components calculate,used together with y pix |
| 9:0 | RW | 0x000 | sw_coe_1st_0 used for all color components calculate,used together with y pix |

VDPU_SWREG2

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:20 | RW | 0x000 | sw_coe_5st used for blue components calculate,used together with cb pix |
| 19:10 | RW | 0x000 | sw_coe_4st used for green color components calculate,used together with cb pix |
| 9:0 | RW | 0x000 | sw_coe_3st used for green color components calculate,used together with cr pix |

VDPU_SWREG3

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | sw_coe_6st used for burrightness adjust,used together with y pix |

VDPU_SWREG4

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:22 | RW | 0x0 | sw_scl_mode_vrt 0 = no scl 1 = up scl 2 = down scl |
| 21:20 | RW | 0x0 | sw_scl_mode_hz 0 = no scl 1 = up scl 2 = down scl |
| 19:18 | RO | 0x0 | reserved |
| 17:0 | RW | 0x00000 | sw_scl_fct_w value = (output_width-1)/(input_width-1) |

VDPU_SWREG5

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:18 | RO | 0x0 | reserved |
| 17:0 | RW | 0x00000 | sw_scl_fct_h value = (output_width-1)/(input_width-1) |

VDPU_SWREG6

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | sw_scl_fct_h_inv value =(inputw-1) / (outputw-1) |
| 15:0 | RW | 0x0000 | sw_scl_fct_w_inv value =(inputw-1) / (outputw-1) |

VDPU_SWREG7

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x0 | reserved |
| 26:16 | RW | 0x000 | sw_pixnum_down_byd Range : 0-dst_height |
| 15:11 | RO | 0x0 | reserved |
| 10:0 | RW | 0x000 | sw_pixnum_up_byd Range : 0-dst_height |

VDPU_SWREG8

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x0 | reserved |
| 26:16 | RW | 0x000 | sw_pixnum_right_byd Range : 0~dst_width |
| 15:11 | RO | 0x0 | reserved |
| 10:0 | RW | 0x000 | sw_pixnum_left_byd Range : 0~dst_width |

VDPU_SWREG9

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | sw_mask_r color R/(alpha channel) component 's bit mask |

VDPU_SWREG10

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | sw_mask_g color G/(alpha channel) component 's bit mask |

VDPU_SWREG11

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | sw_mask_b color B/(alpha channel) component 's bit mask |

VDPU_SWREG12

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | sw_botfld_y_st_adr input bottom field pp start address for y component |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG13

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | sw_botfld_c_st_adr input bottom field pp start address for c component |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG14

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29 | RW | 0x0 | sw_mrrmb_8pix_flag 829PP input picture width is not 16 pixels multiple. Only 8 pixels of the most right MB of the unrotated input picture is used for PP input. |
| 28 | RW | 0x0 | sw_mdmb_8pix_flag the most down unrotated MB of input picture just 8 rows pix data |
| 27 | RO | 0x0 | reserved |
| 26:24 | RW | 0x0 | sw_mbcrop_crdty_ext in order to support jpeg to extend coordinate y bits |
| 23:16 | RW | 0x00 | sw_mbcrop_crdty coordinate y used in macroblock crop |
| 15:12 | RO | 0x0 | reserved |
| 11:9 | RW | 0x0 | sw_mbcrop_crdtx_ext in order to support jpeg to extend bits |
| 8:0 | RW | 0x000 | sw_mbcrop_crdtx coordinate x used in macroblock crop |

VDPU_SWREG15

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x00 | sw_rangemap_coef_c VC- 1:c range map value +9 |
| 7:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | sw_yuv_conv_range Y: 0:16~235 1:0~255 C: 0:16~240 1:0~255 |
| 4:0 | RW | 0x00 | sw_rangemap_y VC- 1:y range map value +9 |

VDPU_SWREG16

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | sw_padd_b the total num of padded in front of B component |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x00 | sw_padd_g the total num of padded in front of G component |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | sw_padd_r the total num of padded in front of R component |

VDPU_SWREG17

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31 | RO | 0x0 | reserved |
| 30 | RO | 0x0 | pp_deinterl_en 0: off 1 : on |
| 29 | RO | 0x0 | pp_abled_en 0: off 1 : on |
| 28 | RO | 0x0 | pp_work_en 0 : off 1: on |
| 27:4 | RO | 0x0 | reserved |
| 3 | RO | 0x0 | pp_outw_1920_en 1st priority used |
| 2 | RO | 0x0 | pp_outw_1280_en 2st priority used |
| 1 | RO | 0x0 | pp_outw_720_en 3st priority used |
| 0 | RO | 0x0 | pp_outw_352_en 4st priority used |

VDPU_SWREG18

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | sw_y_in_st_adr The start address of topfield of the picture when data come from fields.external mode support only |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG19

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | sw_cb_in_st_adr The start address of topfield of the picture when data come from fields.external mode support only |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG20

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | sw_cr_in_st_adr The start address of topfield of the picture when data come from fields,extermal mode support only |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG21

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_y_out_st_adr also the start address of YUYV and RGB |

VDPU_SWREG22

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_c_out_st_adr format is uvuvuv.... |

VDPU_SWREG23

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_abled_st_adr_1st 1.valid when mask1 is used in alpha blending mode 2.Format of data the same as in PP input. 3.Amount of data is related to mask 1 size or ablend1_scanline informed with mask 1 size or with ablend1_scanline if ablend when crop flag valid |

VDPU_SWREG24

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_abled_st_adr_2st 1.valid when mask2 is used in alpha blending mode 2.Format of data the same as in PP input. 3.Amount of data is related to mask 2 size or ablend1_scanline informed with mask 1 size or with ablend1_scanline if ablend when crop flag valid |

VDPU_SWREG25

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:29 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28:16 | RW | 0x0000 | sw_scanl_abld2 corresponding function should be enabled |
| 15:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | sw_scanl_abld1 corresponding function should be enabled |

VDPU_SWREG26

Address: Operational Base + offset (0x0068)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:23 | RO | 0x0 | reserved |
| 22:12 | RW | 0x000 | sw_end_coordx_ma1 range:sw_st_coordx_ma1-dst width |
| 11 | RO | 0x0 | reserved |
| 10:0 | RW | 0x000 | sw_st_coordx_ma1 the start x-coordinate of mask area 1 of Horizontal start pixel |

VDPU_SWREG27

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:23 | RO | 0x0 | reserved |
| 22:12 | RW | 0x000 | sw_end_coordy_ma1 range:sw_st_coordy_ma1~dst width |
| 11 | RO | 0x0 | reserved |
| 10:0 | RW | 0x000 | sw_st_coordy_ma1 the start y-coordinate of mask area 1 of Vertical start pixel |

VDPU_SWREG28

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21:11 | RW | 0x000 | sw_end_coordx_ma2 range:sw_st_coordx_ma2~dst width |
| 10:0 | RW | 0x000 | sw_st_coordx_ma2 the start x-coordinate of mask area 2 of Horizontal start pixel |

VDPU_SWREG29

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21:11 | RW | 0x000 | sw_end_coordy_ma2 range:sw_st_coordy_ma2~dst width |
| 10:0 | RW | 0x000 | sw_st_coordy_ma2 the start y-coordinate of mask area 2 of Vertical start pixel |

VDPU_SWREG30

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | reserved |
| 30:16 | RW | 0x0000 | sw_deinterl_edge Edge detect value used for deinterlacing |
| 15:14 | RO | 0x0 | reserved |
| 13:0 | RW | 0x0000 | sw_deinterl_thr the threshold value of deinterlace |

VDPU_SWREG31

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15:8 | RW | 0x00 | sw_cont_thr1 the threshold value 1 for contrast adjust |
| 7:0 | RW | 0x00 | sw_cont_thr0 the threshold value 0 for contrast adjust |

VDPU_SWREG32

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:26 | RO | 0x0 | reserved |
| 25:16 | RW | 0x000 | sw_cont_offset1 the offset value 1 for contrast adjust |
| 15:10 | RO | 0x0 | reserved |
| 9:0 | RW | 0x000 | sw_cont_offset0 the offset value 0 for contrast adjust |

VDPU_SWREG33

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x1 | abld_crop_flag 0: unsupport crop, the exact image of the area being alpha blended should exist in the external memory 1: support crop, one picture in external memory which come from blended area can be cropped |
| 30 | RO | 0x1 | accut_out_exist_flag PIP: 0 : use 8 pixels (width) or 2 pixels (height) steps to adjust Scaling and masks 1 : use 1 pixel for RGB and 2 pixels for subsampled chroma formats to adjust Scaling and masks |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RO | 0x1 | tile_exist_flag 0 : no exist 1 : exist |
| 28 | RO | 0x1 | dither_exist_flag 0 : no exist 1 : exist |
| 27:26 | RO | 0x3 | scl_perf_sel 0 : without scaling 1 : low performance scaling 2 : high performance scaling 3 : high and fast performance scaling |
| 25 | RO | 0x0 | deinterl_exist_flag 0 : no exist 1 : exist |
| 24 | RO | 0x0 | abld_exist_flag 0 : no exist 1 : exist |
| 23 | RO | 0x1 | pp_in_buf_sel 0 : output buffering is 1 MB 1 : output buffering is 4 MB |
| 22:19 | RO | 0x0 | reserved |
| 18 | RO | 0x1 | pp_endian_mode 0 : Endian mode supported except RGB 1 : Endian mode supported for all format |
| 17 | RO | 0x1 | pp_out_buf_sel 0 : output buffering is 1 unit 1 : output buffering is 4 unit |
| 16 | RO | 0x1 | ppd_exist_flag 0 : no exist 1 : exist |
| 15:14 | RO | 0x1 | pp_tile_in_mode 0 : unsupport 1 : 8x4 tile be used |
| 13:11 | RO | 0x0 | reserved |
| 10:0 | RO | 0x780 | ppd_max_outw the max pixels width allow for pp output |

VDPU_SWREG34

Address: Operational Base + offset (0x0088)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RW | 0x0 | sw_pp_inh_ext in order to support jpeg |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28:21 | RW | 0x00 | sw_pp_inh the picture height of PP input with in macro blocks which can be cropped from a bigger picture when in the condition of external mode |
| 20:12 | RW | 0x000 | sw_org_inw_ext the orginal width of pp input pic in MBS |
| 11:9 | RW | 0x0 | sw_pp_inw_ext in order to support jpeg |
| 8:0 | RW | 0x000 | sw_pp_inw the picture width of PP input with in macro blocks which can be cropped from a bigger picture when in the condition of external mode |

VDPU SWREG35

Address: Operational Base + offset (0x008c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27 | RW | 0x0 | sw_pp_outh_ext the extension height of pp output |
| 26:16 | RW | 0x000 | sw_pp_outh (output width = 2*n (n=1,2,.....) output width =(configuration Pixel Accurate PP output configuration)*n) && (pp output width < 1920 pp output width< 3*(sw_pp_inh-8)) |
| 15:12 | RO | 0x0 | reserved |
| 11 | RW | 0x0 | sw_pp_outw_ext the extension width of pp output |
| 10:0 | RW | 0x000 | sw_pp_outw (output width = 8*n (n=1,2,.....) output width =(configuration Pixel Accurate PP output configuration)*n) && (pp output width < 1920 pp output width< 3*(sw_pp_inw-8)) |

VDPU SWREG36

Address: Operational Base + offset (0x0090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | reserved |
| 5:4 | RW | 0x0 | sw_dither_mode_b 0 : no use dithering 1 : 4-bits dither matrix be used 2 : 5-bits dither matrix be used 3 : 6-bits dither matrix be used |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 3:2 | RW | 0x0 | sw_dither_mode_g 0 : no use dithering 1 : 4-bits dither matrix be used 2 : 5-bits dither matrix be used 3 : 6-bits dither matrix be used |
| 1:0 | RW | 0x0 | sw_dither_mode_r 0 : no use dithering 1 : 4-bits dither matrix be used 2 : 5-bits dither matrix be used 3 : 6-bits dither matrix be used |

VDPU SWREG37

Address: Operational Base + offset (0x0094)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x0 | reserved |
| 26:24 | RW | 0x0 | sw_pp_in_data_strc 0 : Top field 1 : Bottom field 2 : Interlaced field 3 : Interlaced frame 4 : Ripped top field 5 : Ripped bottom field if value=0/1/2,then should read every line from the base address,if value=3/4/5,then should read every second line from the base address |
| 23:20 | RO | 0x0 | reserved |
| 19 | RW | 0x0 | sw_pp_out_crbf_en 0 : Y0CbY0Cr / CbY0CrY0 1 : Y0CrY0Cb / CrY0CbY0 |
| 18 | RW | 0x0 | sw_pp_in_crbf_en yuv422: 0 : Y0CbY0Cr / CbY0CrY0 1 : Y0CrY0Cb / CrY0CbY0 yuv420 semiplanar chrominance: 0 : CbCrCbCr 1:CrCbCrCb |
| 17 | RW | 0x0 | sw_pp_out_yuv_order 0 : Y0CbY0Cr / Y0CrY0Cb 1 : CbY0CrY0 / CrY0CbY0 |
| 16 | RW | 0x0 | sw_pp_in_yuv_order 0 : Y0CbY0Cr / Y0CrY0Cb 1 : CbY0CrY0 / CrY0CbY0 |
| 15:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | sw_pp_out_wordsp it will be used in 64 bit environment 0 : no swapping 1 : swapping high and low 32bit data |
| 10 | RW | 0x0 | sw_pp_out_hfwordsp 0: no swap 1: swap also be used as change pixel orders for 16 bit RGB, support all output format require pp_endian_mode=1 |
| 9 | RW | 0x0 | sw_pp_abld1_in_wordsp for Alpha blend source 1 0 : no swapping 1 : swapping high and low 32bit data |
| 8 | RW | 0x0 | sw_pp_in_wordsp it will be used in 64 bit environment 0 : no swapping 1 : swapping high and low 32bit data |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | sw_rgb_pix_bits 0: every word have only one rga pixel 1: every word have two rga pixel |
| 3 | RW | 0x0 | sw_pp_out_endian for all yuv output endian mode or any data when pp_endian_mode=1 0 : big endian 1 : little endian if pp_endian_mode=0: 16 bit RGB: this bit used as pixel swapping bit 32 bit RGB: no used |
| 2 | RW | 0x0 | sw_pp_abld2_in_endian 0: same with sw_pp_in_endian 1: same with sw_pp_abld1_in_endian '0' = Use PP in endian/swap definitions (sw_pp_in_endian, sw_pp_in_swap) '1' = Use Ablend source 1 endian/swap definitions |
| 1 | RW | 0x0 | sw_pp_abld1_in_endian 0 : big endian (0-1-2-3) 1 : little endian (3-2-1-0) |
| 0 | RW | 0x0 | sw_pp_in_endian this bit will not be used when PP is running pipelined with the decoder 0 : big endian (0-1-2-3) 1 : little endian (3-2-1-0) |

VDPU_SWREG38

Address: Operational Base + offset (0x0098)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:22 | RO | 0x0 | reserved |
| 21:20 | RW | 0x0 | <p>sw_pp_in_tilmod only support yuv420 input data, can be as pipeline or external mode</p> <p>0 : Tiled mode not be activated 1 : 8x4 sized tiles be used 2,3 : reserved</p> |
| 19 | RO | 0x0 | reserved |
| 18:16 | RW | 0x0 | <p>sw_pp_in_fmt_ecp be activated when sw_pp_in_fmt = 3'b111</p> <p>0 : YCbCr 4:4:4 1 : YCbCr 4:1:1</p> |
| 15:14 | RO | 0x0 | reserved |
| 13:11 | RW | 0x0 | <p>sw_pp_out_fmt</p> <p>0 : RGB 1 : YCbCr 4:2:0 ; planar (Not supported) 2 : YCbCr 4:2:2 ; planar (Not supported) 3 : YUYV 4:2:2 ; interleaved 4 : YCbCr 4:4:4 ; planar (Not supported) 5 : YCh 4:2:0 ; chrominance interleaved 6 : YCh 4:2:2 ; (Not supported) 7 : YCh 4:4:4 (Not supported)</p> |
| 10:8 | RW | 0x0 | <p>sw_pp_in_fmt</p> <p>0 : YUYV 4:2:2 ; interleaved and it only supported in external mode 1 : YCbCr 4:2:0 ; the format of Semi-planar in linear raster-scan 2 : YCbCr 4:2:0 ; planar and it only supported in external mode 3 : YCbCr 4:0:0 ; it only supported in pipelined mode 4 : YCbCr 4:2:2 ; Semi-planar and it only supported only in pipelined mode 5 : YCbCr 4:2:0 ; Semi-planar in tiled format and it only supported in external mode 6 : YCbCr 4:4:0 ; Semi-planar and it only supported for jpeg in pipelined mode 7 : same as sw_pp_in_fmt_ecp</p> |
| 7:3 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 2:0 | RW | 0x0 | sw_rot_mode 0 : rotation disabled 1 : rotate + 90 2 : rotate -90 3 : horizontal flip (mirror) 4 : vertical flip 5 : rotate 180 |

VDPU_SWREG39

Address: Operational Base + offset (0x009c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------------------|
| 31:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x000 | sw_display_w Max support 1920 |

VDPU_SWREG40

Address: Operational Base + offset (0x00a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | sw_pp_bus_sts the Interrupt status bit for tell sw bus have some error |
| 2 | RW | 0x0 | sw_pp_rdy_sts the Interrupt status bit for tell sw processed a picture |
| 1 | RW | 0x0 | sw_pp_irq_dis 1 : use polling to see the interrupt 0 : use sw_pp_irq |
| 0 | RW | 0x0 | sw_pp_irq after sw query this interrupt, shoud write 0 to reset. this bit will no used in pipeline mode |

VDPU_SWREG41

Address: Operational Base + offset (0x00a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28 | RW | 0x0 | sw_pp_ahb_hlock_en the service is locked to pp as long as it needs the bus |
| 27 | RW | 0x0 | sw_rightwd_cross_en 0 : disable, 1 : enable |
| 26 | RW | 0x0 | sw_leftsd_cross_en 0 : disable, 1 : enable |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 25 | RW | 0x0 | sw_downwd_cross_en 0 : disable, 1 : enable |
| 24 | RW | 0x0 | sw_upwd_cross_en 0 : disable, 1 : enable |
| 23 | RW | 0x0 | sw_mask2_abld_en alpha blending for the output picture , only be supported when data format is RGB/YUYV422 Alpha blending read data from alpha blend 2 base address |
| 22 | RW | 0x0 | sw_mask1_abld_en alpha blending for the output picture , only be supported when data format is RGB/YUYV422 Alpha blending read data from alpha blend 1 base address |
| 21 | RW | 0x0 | sw_mask2_en 0 : disable, 1 : enable |
| 20 | RW | 0x0 | sw_mask1_en 0 : disable, 1 : enable |
| 19:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | sw_pp_discd_en the burst length will be fix after sw_pp_discd_en=1, and extra read data will auto be discarded by HW |
| 15:12 | RO | 0x0 | reserved |
| 11 | RW | 0x0 | sw_pp_out_tiled_en only used in YCbYCr format . Tile size : 4x4 pixels. |
| 10 | RW | 0x0 | sw_pp_fdscl_en 0 : disable 1 : enabled. it will improve the performance but will decrease the quality of the pic |
| 9 | RW | 0x0 | sw_rangemap_c_en the enable flag for C component Range map |
| 8 | RW | 0x0 | sw_rangemap_y_en VC1: used as range expansion enable |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | sw_pp_pipl_en 0 : disable, external mode 1 : enable, pipeline mode, Post-processing pipeline with decoder |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 3 | RW | 0x1 | <p>sw_pp_clkgate_en pp auto clock gating: default is 1 0 : don't auto gating 1 : auto gating PP dynamic clock gating enable: 1 = Clock is gated from PP structures that are not used 0 = Clock is running for all PP structures Note: Clock gating value can be changed only when PP is not enabled</p> |
| 2 | RW | 0x0 | <p>sw_deint_en the input data should be interlaced format</p> |
| 1 | RW | 0x0 | <p>sw_deint_bld_en on-off Blend for deinterlacing</p> |
| 0 | RW | 0x0 | <p>sw_pp_dec_st after config other register, write 1 to start post-processing operation, and hw will reset to 0 after it decoded a picture should be under External mode.</p> |

VDPU SWREG50

Address: Operational Base + offset (0x00c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:25 | RW | 0x00 | sw_refbuf_pid The used reference picture ID for reference buffer usage |
| 24:13 | RW | 0x000 | sw_refbuf_thrd Used shut down buffer |
| 12 | RW | 0x0 | sw_dec_tiled_lsb 0 : Tiled mode disable 1 : Tiled mode enabled for 8x4 tile size |
| 11 | RW | 0x0 | sw_adv_pref_dis disable for Advanced PREFETCH mode |
| 10 | RW | 0x0 | sw_dec_ascmd0_dis the disable for AXI Single Command Multiple Data0 |
| 9 | RW | 0x0 | sw_skip_sel AVS format: 0 : skip mbs use special MB type 1 : avs skip mbs have the same skip run syntax element as h264 |
| 8 | RW | 0x0 | sw_dblk_flt_dis 1: disable 0: enable |
| 7 | RW | 0x0 | sw_dec_fixed_quant h.264: this bit is for the enable of multi view coding other format(VC1) 0: it can be different inside pic for Quantization parameter 1: it is fixed for Quantization parameter |
| 6:1 | RW | 0x00 | sw_adtion_latency Can be used to slow down 8*sw_dec_latency cycles of IDLE between services, so if sw_dec_latency =0, that is no latency |
| 0 | RW | 0x0 | sw_dec_tiled_msb 0 : Tiled mode disable 1 : Tiled mode enabled for 8x4 tile size |

VDPU SWREG51

Address: Operational Base + offset (0x00cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | reserved |
| 30:25 | RW | 0x00 | sw_qp_init_val the qp(quantization parameter)'s Initial value |
| 24 | RW | 0x0 | sw_strm_len_ext The extension bit of sw_strm_len |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 23:0 | RW | 0x000000 | <p>sw_strm_len if the buffer size be given small than it required, hw will give an interrupt, and then you should config again, and the stream start address should be config also.</p> <p>VC1: one picture/slice of the picture's should be included in the input buffer</p> <p>H264/H263/MPEG*: one slice of the picture's should be included in the input buffer</p> <p>JPEG: 256bytes or onepicture should be included in the input buffer</p> |

VDPU_SWREG52

Address: Operational Base + offset (0x00d0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | reserved |
| 30:17 | RW | 0x0000 | <p>sw_adv_pref_thrd when current MB num > this threshold value, then advanced mode will be closed</p> |
| 16:8 | RW | 0x000 | <p>sw_xdim_mbst it may be used in error concealment case</p> |
| 7:0 | RW | 0x00 | <p>sw_ydim_mbst it may be used in error concealment case</p> |

VDPU_SWREG53

Address: Operational Base + offset (0x00d4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>sw_dec_fmt_sel 0 : H.264, 1 : MPEG-4, 2 : H.263, 3 : JPEG, 4 : VC-1, 5 : MPEG-2, 6 : MPEG-1, 9 : VP7, 11 : AVS, others : reserved</p> |

VDPU_SWREG54

Address: Operational Base + offset (0x00d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | sw_dec_strendian_e 0 : Big endian (0-1-2-3 order) 1 : Little endian (3-2-1-0 order) |
| 4 | RW | 0x0 | sw_dec_strm_wordsp it will be used in 64 bit environment 0 : no swapping 1 : swapping high and low 32bit data |
| 3 | RW | 0x0 | sw_dec_out_wordsp it will be used in 64 bit environment 0 : no swapping 1 : swapping high and low 32bit data |
| 2 | RW | 0x0 | sw_dec_in_wordsp it will be used in 64 bit environment 0 : no swapping 1 : swapping high and low 32bit data note : it no used for stream data |
| 1 | RW | 0x0 | sw_dec_out_endian Decoder output endian mode: 0 : Big endian (0-1-2-3 order) 1 : Little endian (3-2-1-0 order) |
| 0 | RW | 0x0 | sw_dec_in_endian 0 : Big endian (0-1-2-3 order) 1 : Little endian (3-2-1-0 order) note : it no used for stream data |

VDPU_SWREG55

Address: Operational Base + offset (0x00dc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | sw_timeout_det_sts AXI in IDLE status too long |
| 12 | RW | 0x0 | sw_error_det_sts Interrupt status bit input stream error. When high, an error is found in input data stream decoding. HW will self reset. (1,2,3,6,48,55,57) |
| 11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | sw_bslice_det_sts DIVX3:the value extension header flag |
| 9 | RW | 0x0 | sw_slice_det_sts the Interrupt status bit for tell us slice be decoded |
| 8 | RW | 0x0 | sw_aso_det_sts ASO:Arbitrary Slice Ordering |
| 7 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 6 | RW | 0x0 | sw_buf_emt_sts the Interrupt status bit for tell input buffer empty |
| 5 | RW | 0x0 | sw_pp_bus_sts the Interrupt status bit for tell sw bus have some error |
| 4 | RW | 0x0 | sw_dec_rdy_sts the Interrupt status bit for tell sw processed a picture |
| 3:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | sw_dec_irq_dis 1 : use polling to see the interrupt 0 : use sw_pp_irq |
| 0 | RW | 0x0 | sw_dec_irq after sw query this interrupt, shoud write 0 to reset. |

VDPU_SWREG56

Address: Operational Base + offset (0x00e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | sw_axi_sel 0: auto sel for encoder or decoder 1: sel decoder (only used in the middle decoder frame to set bus_dec_en to 0) |
| 22 | RW | 0x0 | sw_dec_data_discd_en the fixed burst length will be used ,and the more read datas will be auto discarded by hw |
| 21 | RW | 0x1 | sw_bus_pos_sel 0:serial 1:parallel |
| 20:16 | RW | 0x00 | sw_dec_max_burlen range : 1-16 |
| 15:8 | RW | 0x01 | sw_dec_axi_id_wr if you config 0,will modify as 1 by hw |
| 7:0 | RW | 0x01 | sw_dec_axi_id_rd if you config 0/5,will modify as 1 by hw |

VDPU_SWREG57

Address: Operational Base + offset (0x00e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | sw_dec_timeout_mode when 1'b0 , timeout cycle is 181'b1 when 1'b1, timeout cycle is 221'b1 |
| 30 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 29 | RW | 0x0 | sw_cache_en 1'b1: cache enable 1'b0: cache disable when sw_cache_en is 1'b1, sw_pref_sigchan should also be 1'b1 |
| 28 | RW | 0x0 | sw_pref_sigchan 1'b1: prefetch single channel enable |
| 27 | RW | 0x0 | sw_intra_dbl3t In chroma dc intra prediction, when this bit is enable, there will 3 cycle enhance for every block |
| 26 | RW | 0x0 | sw_intra_dblspeed Intra double speed enable |
| 25 | RW | 0x0 | sw_inter_dblspeed Inter double speed enable |
| 24:23 | RO | 0x0 | reserved |
| 22 | RW | 0x0 | sw_st_code_exist 0 : not exist 1 : exist |
| 21 | RW | 0x0 | sw_addit_ch_fmt_wen tiled mode should be disable, when this bit be used decoder writes chrominance: group of 8 pixels of Cb then corresponding 8 pixels of Cr Data is written to sw_dec_ch8pix_st_adr. |
| 20 | RW | 0x0 | sw_rlc_mode_en 0:decoder data come from bit stream(VLC mode), side information (bitplane data in VC-1) 1:decoder data come from RLC input data, only h.264 and MPEG4 sp be valid |
| 19 | RW | 0x0 | sw_divx3_en used sw_dec_fmt_sle= MPEG4: 0 : disabled 1 : enabled |
| 18 | RW | 0x0 | sw_prog_jpeg_en 0 : baseline JPEG 1 : progressive JPEG |
| 17 | RW | 0x0 | sw_curpic_code_sel 0 : progressive 1 : interlaced |
| 16 | RW | 0x0 | sw_curpic_stru_sel 0 : frame structure, (that is MBAFF structured picture is interlaced) 1 : field structure |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 15 | RW | 0x0 | sw_pic_type_sel1 0:desided by sw_pic_type_sel0 1: picture type is BI/D/B note: BI is for vc1 D is for mpeg1 B is for h264 |
| 14 | RW | 0x0 | sw_pic_type_sel0 should need sw_pic_type_sel1=0 0: Intra type (I) 1:Inter type (P) |
| 13 | RW | 0x0 | sw_pic_deffield_sel 0 : bottom field 1 : top field |
| 12 | RW | 0x0 | sw_fwd_refpic_mode_sel used for forward reference picture: 0 : progressive 1 : interlaced the backward reference picture is the same as current picture |
| 11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | sw_dmmv_wr_en 0:disable 1:enable this bit used in MPEG2 is for the purpose error concealment case. this bit used in h264 is for the purpose write DPB case with the corresponding reference picture. this bit used in other decoder format is for the purpose writing to external memory starting from mv start address |
| 9 | RW | 0x0 | sw_reftop_en sw_dec_fmt_sel =VC-1 and sw_ref_frm = 0 , 0 = bottom field 1 = top field |
| 8 | RW | 0x0 | sw_first_reftop_en 0 : FWD reference bottom field 1 : FWD reference top field |
| 7 | RW | 0x0 | sw_sequ_mbaff_en 0:disable 1:enable |
| 6 | RW | 0x0 | sw_rd_cnt_tab_en read data from memory used 0:disable 1:enable (hw will read pic order counts) |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 5 | RW | 0x0 | sw_timeout_sts_en 0:disable 1:enable (if hw can be working status too long,you will get an timeout interrupt) |
| 4 | RW | 0x1 | sw_dec_clkgate_en default hw will reset to 1 0:disable 1:enable |
| 3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | sw_dec_wr_extmem_dis 0 : enable 1:disable(no write to external memory) |
| 1 | RW | 0x0 | sw_refpic_buf2_en 0:disable 1:enable (should : pic size > QVGA) |
| 0 | RW | 0x0 | sw_dec_st_work hw will auto reset this be after a frame be decoded no matter it right or have some error |

VDPU SWREG58

Address: Operational Base + offset (0x00e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | sw_soft_rst write 1 to reset,and it will auto reset to 0 after one cycle |

VDPU SWREG59

Address: Operational Base + offset (0x00ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------------------|
| 31:22 | RW | 0x000 | sw_pflt_set0_tap0 0, tap 0 |
| 21:12 | RW | 0x000 | sw_pflt_set0_tap1 set 0, tap 1 |
| 11:2 | RW | 0x000 | sw_pflt_set0_tap2 set 0, tap 2 |
| 1:0 | RO | 0x0 | reserved |

VDPU SWREG60

Address: Operational Base + offset (0x00f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | sw_addit_ch_st_adr The usage is enabled by sw_addit_ch_fmt_wen |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG61

Address: Operational Base + offset (0x00f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | sw_qtable_st_adr JPEG : AC,DC, QP tables MPEG4/2 : QP table H.264 : various tables VP7 : stream decoding tables |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG62

Address: Operational Base + offset (0x00f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | sw_dmmv_st_adr H264: Direct mode motion vector write/read start address Progressive JPEG: the start address for ACDC coefficient read/write If current round is for DC components : this start address is pointing to luminance AC component rounds: this start address is used for current type |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG63

Address: Operational Base + offset (0x00fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | sw_dec_out_st_adr video: write decoder output picture or field start address JPEG snapshot: wirete decoder output luminance picture start address |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG64

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:2 | RW | 0x00000000 | <p>sw_rlc_vlc_st_adr RLC mode: RLC data start address VLC mode: Stream start address HW return value of last_byte_address by this register to tell where stream has been read when you get some abnormality interrupt, may be used for debug VP7: DCT stream for MB rows 0,2n start address</p> |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG65

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31 | RW | 0x0 | <p>sw_refbu_e 0 : disable 1 : enable</p> |
| 30:19 | RW | 0x000 | <p>sw_refbu_thr_level Reference buffer disable threshold value (cache miss amount). Used to buffer shut down (if more misses than allowed)</p> |
| 18:14 | RW | 0x00 | sw_refbuf_picid |
| 13 | RW | 0x0 | <p>sw_refbuf_idcal_e If given threshold level is reached by any picture_id after first MB row, that picture_id is used for reference buffer fill for rest of the picture</p> |
| 12 | RW | 0x0 | <p>sw_refbuf_fildpar_mod_e the mode enable for Field parity mode enable. 0 : the result field of the evaluation be used 1 : the parity mode field be used</p> |
| 11:9 | RO | 0x0 | reserved |
| 8:0 | RW | 0x000 | <p>sw_refbuf_y_ofset if hw should compensate the global motion of the video for better buffer hit rate will use this coordinate</p> |

VDPU_SWREG66

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x6731 | prod_id product number |
| 15:12 | RO | 0x2 | major_num major_num |
| 11:4 | RO | 0x68 | minor_num minor_num |
| 3 | RO | 0x1 | ascii_id_en enable for ASCII product ID |
| 2:0 | RO | 0x0 | build_ver build_ver |

VDPU_SWREG67

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x1 | jpeg_allow_flag 16Mpixel~67Mpixel be sampled and supported by 411 and 444 |
| 30 | RO | 0x1 | refbuf_allow_flag 0: no support 1: support |
| 29 | RO | 0x0 | reserved |
| 28 | RO | 0x0 | refbuf2_allow_flag refbuffer2 support |
| 27:26 | RO | 0x0 | reserved |
| 25 | RO | 0x0 | rom_imp_type 0: from actual ROM units 1: from RTL |
| 24 | RO | 0x1 | vp7_allow_flag vp7 support |
| 23 | RO | 0x0 | reserved |
| 22 | RO | 0x1 | avs_allow_flag avs support |
| 21:20 | RO | 0x1 | mvc_allow_flag mvc support |
| 19 | RO | 0x0 | reserved |
| 18:17 | RO | 0x1 | tile_mode_sel 0: no support 1: 8x4 support 2,3: no used |
| 16:0 | RO | 0x0 | reserved |

VDPU_SWREG68

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0000 | sw_refbuf_sum_top sum of the top partitions |
| 15:0 | RO | 0x0000 | sw_refbuf_sum_bot sum of the bottom partitions |

VDPU_SWREG69

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0000 | sw_refbuf_sum_hit sum of the rebufferd hits of the picture |
| 15:0 | RO | 0x0000 | sw_luma_sum_intra sum of the luminance 8x8 intra partitons of the picture. |

VDPU_SWREG70

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:22 | RO | 0x0 | reserved |
| 21:0 | RO | 0x000000 | sw_ycomp_mv_sum sum of the decoded motion vector y-components |

VDPU_SWREG71

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x1 | dec_mpeg2_allow 0: no support 1: support |
| 30:29 | RO | 0x3 | dec_vc1_allow 0 :not supported 1 :simple profile be supported 2 :main profile be supported 3 :advanced profile be supported |
| 28 | RO | 0x1 | dec_jpeg_allow 0: no support 1: support |
| 27:26 | RO | 0x2 | dec_mpeg4_allow 0 :not supported 1 :simple profile be supported 2 :advanced simple profile be supported |
| 25:24 | RO | 0x3 | dec_h264_allow 0: no support 1: baseline profile be supported 2: high profile be supported |
| 23 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 22 | RO | 0x0 | dec_prog_jpeg_allow 0: no support 1: support |
| 21 | RO | 0x1 | outbuf_sel 0 : 1MB buffer be used 1 : 4MB buffer be used |
| 20 | RO | 0x1 | refbuf_exist 0 : not supported 1 : support |
| 19:16 | RO | 0x5 | dec_std_bus 0 : error 1 : AHB master, AHB slave 2 : OCP master, OCP slave 3 : AXI master, AXI slave 4 : AXI master, APB slave 5 : AXI master, AHB slave |
| 15:14 | RO | 0x1 | rtl_lan_sel 0: no used 1:vhdl 2:verilog |
| 13:12 | RO | 0x2 | bus_w 0 : error 1 : word bus 2 : double word bus 3 : quadruple word bus |
| 11 | RO | 0x0 | reserved |
| 10:0 | RO | 0x780 | sw_dec_max_allow_w the max width can be decoder |

VDPU_SWREG72

Address: Operational Base + offset (0x0120)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30:24 | RO | 0x00 | debug_service this value[6:0]=service_wr[2:0], service_rd[3:0] |
| 23:0 | RO | 0x0 | reserved |

VDPU_SWREG73

Address: Operational Base + offset (0x0124)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | reserved |
| 30 | RO | 0x0 | debug_mv_req mvst_mv_req signal value |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 29 | RO | 0x0 | debug_rlc_req prtr_res_y_req signal value |
| 28 | RO | 0x0 | debug_res_y_req prtr_res_y_req signal value |
| 27 | RO | 0x0 | debug_res_c_req debug_res_c_req |
| 26 | RO | 0x0 | debug_strm_da_e debug_strm_da_e |
| 25 | RO | 0x0 | debug_frm_rdy debug_frm_rdy |
| 24 | RO | 0x0 | debug_flt_req debug_flt_req |
| 23 | RO | 0x0 | debug_ref0_req debug_ref0_req |
| 22 | RO | 0x0 | debug_ref1_req debug_ref1_req |
| 21 | RO | 0x0 | reserved |
| 20:0 | RO | 0x0000000 | debug_mb_cnt debug_mb_cnt |

VDPU SWREG74

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_diff_mv_st_adr Differential motion vector base address used for h264 only it also reuse used as: [29:25] : 9st forward picid of initial reference pic list [24:20] : 8st forward picid of initial reference pic list [19:15] : 7st forward picid of initial reference pic list [14:10] : 6st forward picid of initial reference pic list [9:5] : 5st forward picid of initial reference pic list [4:0] : 4st forward picid of initial reference pic list |
| 1:0 | RO | 0x0 | reserved |

VDPU SWREG75

Address: Operational Base + offset (0x012c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_pred4x4_st_adr also be used as: [29:25] : 15st forward picid of initial reference pic list [24:20] : 14st forward picid of initial reference pic list [19:15] : 13st forward picid of initial reference pic list [14:10] : 12st forward picid of initial reference pic list [9:5] : 11st forward picid of initial reference pic list [4:0] : 10st forward picid of initial reference pic list |
| 1:0 | RO | 0x0 | reserved |

VDPU_SWREG76

Address: Operational Base + offset (0x0130)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx1 the number of reference pic index1 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx0 the number of reference pic index0 |

VDPU_SWREG77

Address: Operational Base + offset (0x0134)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx3 the number of reference pic index3 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx2 the number of reference pic index2 |

VDPU_SWREG78

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx5 the number of reference pic index5 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx4 the number of reference pic index4 |

VDPU_SWREG79

Address: Operational Base + offset (0x013c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx7 the number of reference pic index7 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx6 the number of reference pic index6 |

VDPU_SWREG80

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx9 the number of reference pic index9 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx8 the number of reference pic index8 |

VDPU SWREG81

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx11 the number of reference pic index11 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx10 the number of reference pic index10 |

VDPU SWREG82

Address: Operational Base + offset (0x0148)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx13 the number of reference pic index13 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx12 the number of reference pic index12 |

VDPU SWREG83

Address: Operational Base + offset (0x014c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_num_ref_idx15 the number of reference pic index15 |
| 15:0 | RW | 0x0000 | h264_num_ref_idx14 the number of reference pic index14 |

VDPU SWREG84

Address: Operational Base + offset (0x0150)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref0_st_addr the start address of reference frame0 |
| 1 | RW | 0x0 | h264_ref0_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref0_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU SWREG85

Address: Operational Base + offset (0x0154)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref1_st_addr the start address of reference frame1 |
| 1 | RW | 0x0 | h264_ref1_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref1_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG86

Address: Operational Base + offset (0x0158)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref2_st_addr the start address of reference frame2 |
| 1 | RW | 0x0 | h264_ref2_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref2_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG87

Address: Operational Base + offset (0x015c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref3_st_addr the start address of reference frame3 |
| 1 | RW | 0x0 | h264_ref3_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref3_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG88

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref4_st_addr the start address of reference frame4 |
| 1 | RW | 0x0 | h264_ref4_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref4_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU SWREG89

Address: Operational Base + offset (0x0164)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref5_st_addr the start address of reference frame5 |
| 1 | RW | 0x0 | h264_ref5_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref5_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU SWREG90

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref6_st_addr the start address of reference frame6 |
| 1 | RW | 0x0 | h264_ref6_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref6_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU SWREG91

Address: Operational Base + offset (0x016c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref7_st_addr the start address of reference frame7 |
| 1 | RW | 0x0 | h264_ref7_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref7_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG92

Address: Operational Base + offset (0x0170)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref8_st_addr the start address of reference frame8 |
| 1 | RW | 0x0 | h264_ref8_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref8_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG93

Address: Operational Base + offset (0x0174)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | h264_ref9_st_addr the start address of reference frame9 |
| 1 | RW | 0x0 | h264_ref9_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref9_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG94

Address: Operational Base + offset (0x0178)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | h264_ref10_st_addr the start address of reference frame10 |
| 1 | RW | 0x0 | h264_ref10_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref10_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG95

Address: Operational Base + offset (0x017c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | h264_ref11_st_addr the start address of reference frame11 |
| 1 | RW | 0x0 | h264_ref11_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref11_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU SWREG96

Address: Operational Base + offset (0x0180)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | h264_ref12_st_addr the start address of reference frame12 |
| 1 | RW | 0x0 | h264_ref12_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref12_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU SWREG97

Address: Operational Base + offset (0x0184)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | h264_ref13_st_addr the start address of reference frame13 |
| 1 | RW | 0x0 | h264_ref13_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref13_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU SWREG98

Address: Operational Base + offset (0x0188)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | h264_ref14_st_addr the start address of reference frame14 |
| 1 | RW | 0x0 | h264_ref14_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref14_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG99

Address: Operational Base + offset (0x018c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RW | 0x00000000 | h264_ref15_st_addr the start address of reference frame15 |
| 1 | RW | 0x0 | h264_ref15_field_en 0 : frame 1 : field |
| 0 | RW | 0x0 | h264_ref15_closer_sel 0 : bottom field be selected 1 : top field be selected |

VDPU_SWREG100

Address: Operational Base + offset (0x0190)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29:25 | RW | 0x00 | h264_init_reflist_df5 used for h264 |
| 24:20 | RW | 0x00 | h264_init_reflist_df4 used for h264 |
| 19:15 | RW | 0x00 | h264_init_reflist_df3 used for h264 |
| 14:10 | RW | 0x00 | h264_init_reflist_df2 used for h264 |
| 9:5 | RW | 0x00 | h264_init_reflist_df1 used for h264 |
| 4:0 | RW | 0x00 | h264_init_reflist_df0 used for h264 |

VDPU_SWREG101

Address: Operational Base + offset (0x0194)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:25 | RW | 0x00 | h264_init_reflist_df11 used for h264 |
| 24:20 | RW | 0x00 | h264_init_reflist_df10 used for h264 |
| 19:15 | RW | 0x00 | h264_init_reflist_df9 used for h264 |
| 14:10 | RW | 0x00 | h264_init_reflist_df8 used for h264 |
| 9:5 | RW | 0x00 | h264_init_reflist_df7 used for h264 |
| 4:0 | RW | 0x00 | h264_init_reflist_df6 used for h264 |

VDPU_SWREG102

Address: Operational Base + offset (0x0198)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x00 | h264_init_reflist_df15 used for h264 |
| 14:10 | RW | 0x00 | h264_init_reflist_df14 used for h264 |
| 9:5 | RW | 0x00 | h264_init_reflist_df13 used for h264 |
| 4:0 | RW | 0x00 | h264_init_reflist_df12 used for h264 |

VDPU_SWREG103

Address: Operational Base + offset (0x019c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29:25 | RW | 0x00 | h264_init_reflist_db5 used for h264 |
| 24:20 | RW | 0x00 | h264_init_reflist_db4 used for h264 |
| 19:15 | RW | 0x00 | h264_init_reflist_db3 used for h264 |
| 14:10 | RW | 0x00 | h264_init_reflist_db2 used for h264 |
| 9:5 | RW | 0x00 | h264_init_reflist_db1 used for h264 |
| 4:0 | RW | 0x00 | h264_init_reflist_db0 used for h264 |

VDPU_SWREG104

Address: Operational Base + offset (0x01a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:25 | RW | 0x00 | h264_init_reflist_db11 used for h264 |
| 24:20 | RW | 0x00 | h264_init_reflist_db10 used for h264 |
| 19:15 | RW | 0x00 | h264_init_reflist_db9 used for h264 |
| 14:10 | RW | 0x00 | h264_init_reflist_db8 used for h264 |
| 9:5 | RW | 0x00 | h264_init_reflist_db7 used for h264 |
| 4:0 | RW | 0x00 | h264_init_reflist_db6 used for h264 |

VDPU_SWREG105

Address: Operational Base + offset (0x01a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x00 | h264_init_reflist_db15 used for h264 |
| 14:10 | RW | 0x00 | h264_init_reflist_db14 used for h264 |
| 9:5 | RW | 0x00 | h264_init_reflist_db13 used for h264 |
| 4:0 | RW | 0x00 | h264_init_reflist_db12 used for h264 |

VDPU_SWREG106

Address: Operational Base + offset (0x01a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x00 | h264_init_reflist_pf3 Initial reference picture list for P forward picid 3 |
| 14:10 | RW | 0x00 | h264_init_reflist_pf2 Initial reference picture list for P forward picid 2 |
| 9:5 | RW | 0x00 | h264_init_reflist_pf1 Initial reference picture list for P forward picid 1 |
| 4:0 | RW | 0x00 | h264_init_reflist_pf0 used in 264 |

VDPU_SWREG107

Address: Operational Base + offset (0x01ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | h264_refpic_term_flag long term flag for reference picture index |

VDPU_SWREG108

Address: Operational Base + offset (0x01b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | h264_refpic_valid_flag valid flag for reference picture index |

VDPU_SWREG109

Address: Operational Base + offset (0x01b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x00 | h264_strm_start_bit associates with sw_rlc_vlc_st_adr |

VDPU_SWREG110

Address: Operational Base + offset (0x01b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x0 | reserved |
| 26:22 | RW | 0x00 | h264_flt_offset_cr_qp filter offset of cr qp |
| 21:17 | RW | 0x00 | h264_flt_offset_cb_qp filter offset of cb qp |
| 16:9 | RW | 0x00 | h264_pic_mb_h value =((pixel height+15)/16). used for frame or single field size being decoded |
| 8:0 | RW | 0x000 | h264_pic_mb_w value = ((pixel width + 15) /16) |

VDPU_SWREG111

Address: Operational Base + offset (0x01bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:18 | RO | 0x0 | reserved |
| 17:16 | RW | 0x0 | h264_wp_bslice_sel 0 : default wp be used 1 : explicit wp be used 2 : implicit wp be used |
| 15:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | h264_max_refnum this value is for decoded picture buffer |

VDPU_SWREG112

Address: Operational Base + offset (0x01c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | h264_dblk_ctrl_flag to indicates if the slice header will have the deblocking filter's extra variables controlling characteristics |
| 30 | RW | 0x0 | h264_rpcp_flag to specifies whether redundant picture count syntax elements |
| 29:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | h264_curfrm_len H.264: Bit length of frame_num in data stream |
| 15:0 | RW | 0x0000 | h264_curfrm_num it may be use for reference picture reordering and identify short-term reference frames |

VDPU SWREG113

Address: Operational Base + offset (0x01c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:27 | RO | 0x0 | reserved |
| 26:16 | RW | 0x000 | h264_mk_len use for decoded reference picture |
| 15:0 | RW | 0x0000 | h264_idrp_id instantaneous decoding refresh picture id |

VDPU SWREG114

Address: Operational Base + offset (0x01c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0x00 | h264_pps_id it identifies the slice header will have the id of picture parameter set |
| 23:19 | RW | 0x00 | h264_max_refidx1 it will be used in decoding inter predicted macro blocks |
| 18:14 | RW | 0x00 | h264_max_refidx0 it will be used in decoding inter predicted macro blocks |
| 13:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | h264_pocf_len the length of picture order count field in stream |

VDPU SWREG115

Address: Operational Base + offset (0x01cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | h264_idr_pic_flag instantaneous decoding refresh picture flag |
| 7 | RW | 0x0 | h264_dlmv_method_en with B_skip, B_Direct_16x16 and B_direct_8x8_inference_flag |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 6 | RW | 0x0 | h264_monochr_en sampling format , 0 : 4:2:0 1 : 4:0:0 |
| 5 | RW | 0x0 | h264_cabac_en enable for cabac |
| 4 | RW | 0x0 | h264_pslice_wp_en enable flag of Weighted prediction for P slices |
| 3 | RW | 0x0 | h264_nimb_intra_en 0 : neighbouring inter macroblocks are used in intra prediction process 1 : neighbouring intra macroblocks are used |
| 2 | RW | 0x0 | h264_trnff_flag_en_8x8 8x8 transform flag enable for stream decoding |
| 1 | RW | 0x0 | h264_scl_matrix_en 0 : normal transform 1 : use scaling matrix for transform |
| 0 | RW | 0x0 | h264_fieldpic_flag_exist Flag for streamd that field_pic_flag exists in stream |

VDPU_SWREG120

Address: Operational Base + offset (0x01e0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg0 MPEG4/JPEG/VC-1/MPEG2/VP6/VP7/VP: [31:2] RLC mode: Base address for RLC data VLC mode: Stream start address VP7: [31:2] This base address is used as sw_dct_strm0_base including DCT stream for MB rows 0,2n |

VDPU_SWREG121

Address: Operational Base + offset (0x01e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg1</p> <p>vc-1:</p> <ul style="list-style-type: none"> [31] : bitplane mode enable for corresponding element0 [30] : bitplane mode enable for corresponding element1 [29] : bitplane mode enable for corresponding element2 [28:24] : alemative PQUANT. [23:20] : weather dq_profile will be set to picture edges [19] : TTMB or TTFRM sel [18:14] : qpindex value [12] : enable for bilinear motion compensation ena [11] : enable for iform quantizer e [10] : enable for HALFQP [9:8] : frame level transform type sel [7] : the 2st byte of the stream emulation byte [6] : enable for antization parameter change inside frame [5] : enable for VC-1 advanced profile <p>JPEG:</p> <ul style="list-style-type: none"> [26:0] : progressive JPEG <p>MPEG2 :</p> <ul style="list-style-type: none"> [12] : enable for bilinear motion compensation <p>VP7:</p> <ul style="list-style-type: none"> [31:26] : DCT stream partition index 1 of start bit [25:20] : DCT stream partition index 2 of start bit [13] : rominance motion vector resolution for VP7/8 [12] : enable for bilinear motion compensation [11:9] : 0st count for DC prediction mach [8:6] : 1st count for DC prediction mach |

VDPU_SWREG122

Address: Operational Base + offset (0x01e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg2</p> <p>MPEG4:</p> <ul style="list-style-type: none"> [31:26] : Exact bit of stream start word [25] : enable for sync markers [24] : enable for Type 1 quantization [23:19] : the offset of Qp filter [18:14] : the offset of Qp filter for cr [0] : filed_pic_flag exists in stream <p>JPEG :</p> <ul style="list-style-type: none"> [31:26] : Exact bit of stream start word [25] : enable for sync markers [12:11] : total of Quantization tables [10:8] : the sampling format for input pic [7] : JPEG width [6] : weather current strem buffer contain the end of a JPEG image [5:0] : vlc table <p>VC-1 :</p> <ul style="list-style-type: none"> [31:26] : Exact bit of stream start word [25] : enable for sync markers [24] : Quantisation profile [23] : each mb take on quantization step size or not [22] : range reduced [20] : chrominance interpolation accuracy information [17:13] : select tables which be used to dec [12:10] : select mode syntax element table [9:7] : select mv table [6:4] : select CPBCY table [3:0] : block pattern table select <p>vp6:</p> <ul style="list-style-type: none"> [23:18] : start bit for ctrl stream (vp7) [17] : enable for huffman decoding [16] : enable for muti stream (vp7) [15:8] : boolean dec init value(vp7) [7:0] : boolean dec init range |

VDPU_SWREG123

Address: Operational Base + offset (0x01ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg3</p> <p>JPEG:</p> <ul style="list-style-type: none"> [15:0] : start marker frequency. <p>vc-1:</p> <ul style="list-style-type: none"> [31:24] : B picture scl factor [23:19] : FWD direction reference distance [18:14] : BWD direction reference distance <p>vp6:</p> <ul style="list-style-type: none"> [17:14] : loop filter limit value [13] : enable for variance test [12:10] : filter MV size threshold [9:0] : filter variance threshold <p>VP7/VP :</p> <ul style="list-style-type: none"> [31:16] : value 0 for initial dc predictor [15:0] : value 1 for initial dc predictor |

VDPU_SWREG124

Address: Operational Base + offset (0x01f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg4</p> <p>MPEG4:</p> <ul style="list-style-type: none"> [31:2] : MB ctrl start address <p>VC-1:</p> <ul style="list-style-type: none"> [24] : enable for 0st intensity compensation [23:16] : iscale value [15:0] : ishift value <p>VP6:</p> <ul style="list-style-type: none"> [23:0] : total of CTRL stream data <p>VP7/VP :</p> <ul style="list-style-type: none"> [27:24] : coefficient partitions number [23:0] : total of CTRL stream data |

VDPU_SWREG125

Address: Operational Base + offset (0x01f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg5</p> <p>JPEG:</p> <ul style="list-style-type: none"> [31:2] : Cb ACDC coeff start address <p>VP6/VP7/vp:</p> <ul style="list-style-type: none"> [31:22] : prediction filter with set 5 and tap 1 [21:12] : prediction filter with set 5 and tap 2 [11:2] : prediction filter with set 5 and tap 3 |

VDPU_SWREG126

Address: Operational Base + offset (0x01f8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg6 JPEG: [31:2] : Cr ACDC coeff start address VP6/VP7/vp: [31:22] : prediction filter with set 6 and tap 0 [21:12] : prediction filter with set 6 and tap 1 [11:2] : prediction filter with set 6 and tap 2 |

VDPU_SWREG127

Address: Operational Base + offset (0x01fc)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg7 VP6/VP7/vp: [31:22] : prediction filter with set 6 and tap 3 [21:12] : prediction filter with set 7 and tap 0 [11:2] : prediction filter with set 7 and tap 1 |

VDPU_SWREG128

Address: Operational Base + offset (0x0200)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg8 VP6: [31:22] : prediction filter with set 7 and tap 2 [21:12] : prediction filter with set 7 and tap 3 VP7/VP: [31:22] : prediction filter with set 7 and tap 2 [21:12] : prediction filter with set 7 and tap 3 [11:10] : extra prediction filter with set 2 and tap -1 [9:8] : extra prediction filter with set 2 and tap 4 [7:6] : extra prediction filter with set 4 and tap -1 [5:4] : extra prediction filter with set 4 and tap 4 [3:2] : extra prediction filter with set 6 and tap -1 [1:0] : extra prediction filter with set 6 and tap 4 |

VDPU_SWREG129

Address: Operational Base + offset (0x0204)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg9 VP6: [29:24] : 56st coef of scan read index [23:18] : 57st coef of scan read index [17:12] : 58st coef of scan read index [11:6] : 59st coef of scan read index [5:0] : 60st coef of scan read index |

VDPU_SWREG130

Address: Operational Base + offset (0x0208)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg10 VP6: [29:24] : 61st coef of scan read index [23:18] : 62st coef of scan read index [17:12] : 63st coef of scan read index VP7: [21:11] : QP0 for VP7 [10:0] : QP1 for VP7 |

VDPU_SWREG131

Address: Operational Base + offset (0x020c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg11 MPEG4/H263/VC-1/vp6/VP7: [31:2] : reference pic0 start address JPEG: [31:2] : the ch decoder output start address |

VDPU_SWREG132

Address: Operational Base + offset (0x0210)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg12 VP7: [31] : type of loop filter [30:28] : sharpness of loop filter [27:21] : MB type0 adjustment of filter level [20:14] : MB type1 adjustment of filter level [13:7] : MB type2 adjustment of filter level [6:0] : MB type3 adjustment of filter level |

VDPU_SWREG133

Address: Operational Base + offset (0x0214)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg13</p> <p>VP6:</p> <ul style="list-style-type: none"> [29:24] : 51st coef of scan read index [23:18] : 52st coef of scan read index [17:12] : 53st coef of scan read index [11:6] : 54st coef of scan read index [5:0] : 55st coef of scan read index <p>VP7:</p> <ul style="list-style-type: none"> [27:21] : reference frame type0 adjustment of filter level [20:14] : reference frame type1 adjustment of filter level [13:7] : reference frame type2 adjustment of filter level [6:0] : reference frame type3 adjustment of filter level |

VDPU_SWREG134

Address: Operational Base + offset (0x0218)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg14</p> <p>MPEG4/vc1/MEPG2:</p> <ul style="list-style-type: none"> [31:2] : reference pic2 start address <p>JPEG:</p> <ul style="list-style-type: none"> [30:24] : code words of length 6 [21:16] : code words of length 5 [15:11] : code words of length 4 [10:7] : code words of length 3 [5:3] : code words of length 2 [1:0] : code words of length 1 |

VDPU_SWREG135

Address: Operational Base + offset (0x021c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg15</p> <p>MPEG4/vc1/MEPG2:</p> <ul style="list-style-type: none"> [31:2] : reference pic3 start address <p>JPEG:</p> <ul style="list-style-type: none"> [30:24] : code words of length 10 [23:16] : code words of length 9 [15:8] : code words of length 8 [7:0] : code words of length 7 |

VDPU_SWREG136

Address: Operational Base + offset (0x0220)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg16</p> <p>VP6/VP7</p> <p>[31:2] : golden reference pic start address(PIC_ID 4)</p> <p>[0] : golden reference pic siggn bias(VP7/VP8)</p> <p>VC-1:</p> <p>[31:16] : length of picture header</p> <p>[13] : 1/4 MV/MB sel</p> <p>[11] : enable for ref pic range reduce</p> <p>[10:9] : max different mv length</p> <p>[7:6] : select range for mv</p> <p>[5] : enable for overlap smoothing</p> <p>[4:3] : MB overlap smoothing method</p> <p>MPEG4/MPEG2:</p> <p>[19] : alternalte scan flag</p> <p>[18:15] : HRZ AXI's bit amount for representing FWD MV</p> <p>[14:11] : VRZ AXI's bit amount for representing FWD MV</p> <p>[10:7] : HRZ AXI's bit amount for representing BWD MV</p> <p>[6:3] : VRZ AXI's bit amount for representing BWD MV</p> <p>[2] : FWD MV Y resolution</p> <p>[1] : the ctrl bit for rounding(MPEG4),BWD MV Y resolution(MPEG2)</p> <p>[0] : pic type of previous anchor(MPEG4)</p> <p>JPEG:</p> <p>[30:24] : code words of length 14</p> <p>[23:16] : code words of length 13</p> <p>[15:8] : code words of length 12</p> <p>[7:0] : code words of length 11</p> |

VDPU_SWREG137

Address: Operational Base + offset (0x0224)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg17</p> <p>MPEG4:</p> <ul style="list-style-type: none"> [26:0] : reference distance syntax for delta value0 be used <p>JPEG:</p> <ul style="list-style-type: none"> [31:27] : tab2:code words of length 4 [26:23] : tab2:code words of length 3 [21:19] : tab2:code words of length 2 [17:16] : tab2:code words of length 1 [15:8] : tab1:code words of length 16 [7:0] : tab1:code words of length 15 <p>VC-1:</p> <ul style="list-style-type: none"> [24] : enable for intensity compensation 3 [23:16] : intensity compensation's iscale value [15:0] : intensity compensation's ishift value <p>VP6/VP7:</p> <ul style="list-style-type: none"> [29:24] : 1st coef of scan read index [23:18] : 2st coef of scan read index [17:12] : 3st coef of scan read index [11:6] : 4st coef of scan read index [5:0] : 5st coef of scan read index |

VDPU_SWREG138

Address: Operational Base + offset (0x0228)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg18</p> <p>MPEG4:</p> <ul style="list-style-type: none"> [26:0] : reference distance syntax for delta value -1 be used <p>JPEG:</p> <ul style="list-style-type: none"> [31:24] : tab2:code words of length 8 [23:16] : tab2:code words of length 7 [14:8] : tab2:code words of length 6 [5:0] : tab2:code words of length 5 <p>VC-1:</p> <ul style="list-style-type: none"> [24] : enable for intensity compensation 4 [23:16] : intensity compensation's iscale value [15:0] : intensity compensation's shift value <p>VP6/VP7:</p> <ul style="list-style-type: none"> [29:24] : 6st coef of scan read index [23:18] : 7st coef of scan read index [17:12] : 8st coef of scan read index [11:6] : 9st coef of scan read index [5:0] : 10st coef of scan read index |

VDPU_SWREG139

Address: Operational Base + offset (0x022c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg19</p> <p>MPEG4:</p> <ul style="list-style-type: none"> [26:0] : reference distance syntax for delta value1 be used <p>JPEG:</p> <ul style="list-style-type: none"> [31:24] : tab2:code words of length 12 [23:16] : tab2:code words of length 11 [15:8] : tab2:code words of length 10 [7:0] : tab2:code words of length 9 <p>VP6/VP7:</p> <ul style="list-style-type: none"> [29:24] : 11st coef of scan read index [23:18] : 12st coef of scan read index [17:12] : 13st coef of scan read index [11:6] : 14st coef of scan read index [5:0] : 15st coef of scan read index |

VDPU_SWREG140

Address: Operational Base + offset (0x0230)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg20</p> <p>JPEG:</p> <ul style="list-style-type: none"> [31:24] : tab2:code words of length 16 [23:16] : tab2:code words of length 15 [15:8] : tab2:code words of length 14 [7:0] : tab2:code words of length 13 <p>VP6:</p> <ul style="list-style-type: none"> [29:24] : 16st coef of scan read index [23:18] : 17st coef of scan read index [17:12] : 18st coef of scan read index [11:6] : 19st coef of scan read index [5:0] : 20st coef of scan read index <p>VP7:</p> <ul style="list-style-type: none"> [31:2] : DCT stream MB row 2,2n+1 start address |

VDPU_SWREG141

Address: Operational Base + offset (0x0234)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg21</p> <p>JPEG:</p> <ul style="list-style-type: none"> [31:28] : tab1:code words of length 8 [27:24] : tab1:code words of length 7 [23:20] : tab1:code words of length 6 [19:16] : tab1:code words of length 5 [15:12] : tab1:code words of length 4 [11:8] : tab1:code words of length 3 [6:4] : tab1:code words of length 2 [1:0] : tab1:code words of length 1 <p>VP6:</p> <ul style="list-style-type: none"> [29:24] : 21st coef of scan read index [23:18] : 22st coef of scan read index [17:12] : 23st coef of scan read index [11:6] : 24st coef of scan read index [5:0] : 25st coef of scan read index <p>VP7</p> <ul style="list-style-type: none"> [31:2] : DCT stream MB row 2,2n+2 start address |

VDPU SWREG142

Address: Operational Base + offset (0x0238)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg22</p> <p>JPEG:</p> <ul style="list-style-type: none"> [31:28] : tab1:code words of length 16 [27:24] : tab1:code words of length 15 [23:20] : tab1:code words of length 14 [19:16] : tab1:code words of length 13 [15:12] : tab1:code words of length 12 [11:8] : tab1:code words of length 11 [6:4] : tab1:code words of length 10 [1:0] : tab1:code words of length 9 <p>VP6:</p> <ul style="list-style-type: none"> [29:24] : 26st coef of scan read index [23:18] : 27st coef of scan read index [17:12] : 28st coef of scan read index [11:6] : 29st coef of scan read index [5:0] : 30st coef of scan read index <p>VP7</p> <ul style="list-style-type: none"> [31:2] : DCT stream MB row 2,2n+3 start address |

VDPU_SWREG143

Address: Operational Base + offset (0x023c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>mfr_reg23</p> <p>JPEG:</p> <ul style="list-style-type: none"> [31:28] : tab2:code words of length 8 [27:24] : tab2:code words of length 7 [23:20] : tab2:code words of length 6 [19:16] : tab2:code words of length 5 [15:12] : tab2:code words of length 4 [11:8] : tab2:code words of length 3 [6:4] : tab2:code words of length 2 [1:0] : tab2:code words of length 1 <p>VP6:</p> <ul style="list-style-type: none"> [29:24] : 31st coef of scan read index [23:18] : 32st coef of scan read index [17:12] : 33st coef of scan read index [11:6] : 34st coef of scan read index [5:0] : 35st coef of scan read index <p>VP7</p> <ul style="list-style-type: none"> [31:2] : DCT stream MB row 2,2n+4 start address |

VDPU_SWREG144

Address: Operational Base + offset (0x0240)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>mfr_reg24</p> <p>JPEG:</p> <ul style="list-style-type: none"> [31:28] : tab2:code words of length 16 [27:24] : tab2:code words of length 15 [23:20] : tab2:code words of length 14 [19:16] : tab2:code words of length 13 [15:12] : tab2:code words of length 12 [11:8] : tab2:code words of length 11 [6:4] : tab2:code words of length 10 [1:0] : tab2:code words of length 9 <p>VP6:</p> <ul style="list-style-type: none"> [29:24] : 36st coef of scan read index [23:18] : 37st coef of scan read index [17:12] : 38st coef of scan read index [11:6] : 39st coef of scan read index [5:0] : 40st coef of scan read index <p>VP7</p> <ul style="list-style-type: none"> [31:2] : DCT stream MB row 2,2n+5 start address |

VDPU_SWREG145

Address: Operational Base + offset (0x0244)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg25 JPEG: [31:28] : tab3:code words of length 8 [27:24] : tab3:code words of length 7 [23:20] : tab3:code words of length 6 [19:16] : tab3:code words of length 5 [15:12] : tab3:code words of length 4 [11:8] : tab3:code words of length 3 [6:4] : tab2:code words of length 2 [1:0] : tab3:code words of length 1 VC-1: [31:2] : bitplane mb ctrl start address VP6/VP7: [31:2] : ctrl data stream start address |

VDPU_SWREG146

Address: Operational Base + offset (0x0248)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg26 JPEG: [31:28] : tab3:code words of length 16 [27:24] : tab3:code words of length 15 [23:20] : tab3:code words of length 14 [19:16] : tab3:code words of length 13 [15:12] : tab3:code words of length 12 [11:8] : tab3:code words of length 11 [6:4] : tab3:code words of length 10 [1:0] : tab3:code words of length 9 VP6: [29:24] : 41st coef of scan read index [23:18] : 42st coef of scan read index [17:12] : 43st coef of scan read index [11:6] : 44st coef of scan read index [5:0] : 45st coef of scan read index VP7 [31:2] : DCT stream MB row 2,2n+6 start address |

VDPU_SWREG147

Address: Operational Base + offset (0x024c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg27 VP6: [29:24] : 46st coef of scan read index [23:18] : 47st coef of scan read index [17:12] : 48st coef of scan read index [11:6] : 49st coef of scan read index [5:0] : 50st coef of scan read index VP7: [31:2] : DCT stream MB row 2,2n+7 start address |

VDPU_SWREG148

Address: Operational Base + offset (0x0250)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg28 MPEG4/VC-1/MPEG2/VP7: [31:2] : ref pic index 1 start address JPEG: [7:0] : snapshot |

VDPU_SWREG149

Address: Operational Base + offset (0x0254)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg29 VC-1: [24] : enable for intensity compensation 1 [23:16] : intensity compensation iscale value [15:0] : intensity compensation ishift value VP7: [31:2] : the segmentation map value start address [1] : enable for segmentation map update [0] : enable for segmentation |

VDPU_SWREG150

Address: Operational Base + offset (0x0258)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg30 VC-1: [24] : enable for intensity compensation 2 [23:16] : intensity compensation iscale value [15:0] : intensity compensation ishift value VP7: [29:24] : DCT stream partition index 3 of start bit [23:18] : DCT stream partition index 4 of start bit [17:12] : DCT stream partition index 5 of start bit [11:6] : DCT stream partition index 6 of start bit [5:0] : DCT stream partition index 7 of start bit |

VDPU_SWREG151

Address: Operational Base + offset (0x025c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg31 VP7: [21:11] : QP2 for VP7 [10:0] : QP3 for VP7 |

VDPU_SWREG152

Address: Operational Base + offset (0x0260)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg32 VP7: [21:11] : QP4 for VP7 [10:0] : QP5 for VP7 |

VDPU_SWREG153

Address: Operational Base + offset (0x0264)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg33 VC-1/VP6/VP7: [31:22] : prediction filter with set 0,tap3 (also for mpeg4) [21:12] : prediction filter with set 1,tap0 [11:2] : prediction filter with set 1,tap1 |

VDPU_SWREG154

Address: Operational Base + offset (0x0268)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg34 VC-1/VP6/VP7: [31:22] : prediction filter with set 1,tap2 [21:12] : prediction filter with set 1,tap3 [11:2] : prediction filter with set 2,tap0(no for vc-1) |

VDPU_SWREG155

Address: Operational Base + offset (0x026c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | mfr_reg35 VC-1/VP6/VP7: [31:22] : prediction filter with set 2,tap1 [21:12] : prediction filter with set 2,tap2 [11:2] : prediction filter with set 2,tap3 |

VDPU_SWREG156

Address: Operational Base + offset (0x0270)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg36 VP6/VP7: [31:22] : prediction filter with set 3,tap0 [21:12] : prediction filter with set 3,tap1 [11:2] : prediction filter with set 3,tap2 |

VDPU_SWREG157

Address: Operational Base + offset (0x0274)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg37 VP6/VP7: [31:22] : prediction filter with set 3,tap3 [21:12] : prediction filter with set 4,tap0 [11:2] : prediction filter with set 4,tap1 |

VDPU_SWREG158

Address: Operational Base + offset (0x0278)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | mfr_reg38 VP6/VP7: [31:22] : prediction filter with set 4,tap2 [21:12] : prediction filter with set 4,tap3 [11:2] : prediction filter with set 5,tap0 |

VDPU_SWREG164 PERF LATENCY CTRL0

Address: Operational Base + offset (0x0290)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:8 | RW | 0x000 | sw_rd_latency_thr |
| 7:4 | RW | 0x0 | sw_rd_latency_id |
| 3 | RW | 0x0 | sw_axi_cnt_type sw_axi_cnt_type |
| 2 | RW | 0x0 | sw_axi_perf_frm_type 1'b0: clear by frame end 1'b1: clear by software configuration |
| 1 | W1C | 0x0 | sw_axi_perf_clr_e 1'b0: software clear disable 1'b1: software clear enable |
| 0 | RW | 0x0 | sw_axi_perf_work_e 1'b0: disable 1'b1: enable |

VDPU_SWREG165_PERF_LATENCY_CTRL1

Address: Operational Base + offset (0x0294)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x0 | sw_aw_count_id sw_aw_count_id |
| 7:4 | RW | 0x0 | sw_ar_count_id sw_ar_count_id |
| 3 | RW | 0x0 | sw_aw_cnt_id_type sw_aw_cnt_id_type |
| 2 | RW | 0x0 | sw_ar_cnt_id_type sw_ar_cnt_id_type |
| 1:0 | RW | 0x0 | sw_addr_align_type sw_addr_align_type |

VDPU_SWREG166_PERF_RD_MAX_LATENCY_NUM0

Address: Operational Base + offset (0x0298)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:16 | RO | 0x0 | reserved |
| 15:0 | RO | 0x0000 | rd_max_latency_num_ch0 |

VDPU_SWREG167_PERF_RD_LATENCY_SAMP_NUM

Address: Operational Base + offset (0x029c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:0 | RO | 0x00000000 | rd_latency_thr_num_ch0 |

VDPU_SWREG168_PERF_RD_LATENCY_ACC_SUM

Address: Operational Base + offset (0x02a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:0 | RO | 0x00000000 | rd_latency_acc_sum |

VDPU_SWREG169_PERF_RD_AXI_TOTAL_BYTE

Address: Operational Base + offset (0x02a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | perf_rd_axi_total_byte perf_rd_axi_total_byte |

VDPU_SWREG170_PERF_WR_AXI_TOTAL_BYTE

Address: Operational Base + offset (0x02a8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | perf_wr_axi_total_byte perf_wr_axi_total_byte |

VDPU SWREG171 PERF WORKING CNT

Address: Operational Base + offset (0x02ac)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--------------------------------------|
| 31:0 | RW | 0x00000000 | perf_working_cnt perf_working_cnt |

4.5.4 VDPU121 MMU Register Summary

| Name | Offset | Size | Reset Value | Description |
|-----------------------------------|--------|------|-------------|---|
| <u>VCODEC_MMU_DTE_ADDR</u> | 0x0000 | W | 0x00000000 | MMU current page Table addressIt is only can be written when MMU state is disable or page fault or mmu enable stall state |
| <u>VCODEC_MMU_STATUS</u> | 0x0004 | W | 0x00000018 | MMU status register |
| <u>VCODEC_MMU_COMMAND</u> | 0x0008 | W | 0x00000000 | MMU command register |
| <u>VCODEC_MMU_PAGE_FAULT_ADDR</u> | 0x000c | W | 0x00000000 | MMU logical address of last page fault |
| <u>VCODEC_MMU_ZAP_ONE_LINE</u> | 0x0010 | W | 0x00000000 | MMU Zap cache line register |
| <u>VCODEC_MMU_INT_RAW_STAT</u> | 0x0014 | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_INT_CLEA_R</u> | 0x0018 | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_INT_MASK</u> | 0x001c | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_INT_STATUS</u> | 0x0020 | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_AUTO_GATING</u> | 0x0024 | W | 0x00000001 | mmu auto gating |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access**4.5.5 VDPU121 MMU Detail Register Description****VCODEC_MMU_DTE_ADDR**

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | MMU_DTE_ADDR MMU current page Table address |

VCODEC_MMU_STATUS

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:11 | RO | 0x0 | reserved |
| 10:6 | RO | 0x00 | PAGE_FAULT_BUS_ID Index of master responsible for last page fault |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 5 | RO | 0x0 | PAGE_FAULT_IS_WRITE The direction of access for last page fault: 0 = Read 1 = Write |
| 4 | RO | 0x1 | REPLAY_BUFFER_EMPTY 1'b1: The MMU replay buffer is empty |
| 3 | RO | 0x1 | MMU_IDLE The MMU is idle when accesses are being translated and there are no unfinished translated accesses. 1'b1: MMU is idle |
| 2 | RO | 0x0 | STAIL_ACTIVE MMU stall mode currently enabled. The mode is enabled by command 1'b1: MMU is in stall active status |
| 1 | RO | 0x0 | PAGE_FAULT_ACTIVE MMU page fault mode currently enabled . The mode is enabled by command. 1'b1: page fault is active |
| 0 | RO | 0x0 | PAGING_ENABLED 1'b0: paging is disabled 1'b1: Paging is enabled |

VCODEC_MMU_COMMAND

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:3 | RO | 0x0 | reserved |
| 2:0 | WO | 0x0 | MMU_CMD MMU_CMD. This can be: 0: MMU_ENABLE_PAGING 1: MMU_DISABLE_PAGING 2: MMU_ENABLE_STALL 3: MMU_DISABLE_STALL 4: MMU_ZAP_CACHE 5: MMU_PAGE_FAULT_DONE 6: MMU_FORCE_RESET |

VCODEC_MMU_PAGE_FAULT_ADDR

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | PAGE_FAULT_ADDR address of last page fault |

VCODEC_MMU_ZAP_ONE_LINE

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | WO | 0x00000000 | MMU_ZAP_ONE_LINE address to be invalidated from the page table cache |

VCODEC_MMU_INT_RAWSTAT

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR read bus error status |
| 0 | RW | 0x0 | PAGE_FAULT page fault status |

VCODEC_MMU_INT_CLEAR

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | WO | 0x0 | READ_BUS_ERROR write 1 to clear read bus error |
| 0 | WO | 0x0 | PAGE_FAULT write 1 to page fault clear |

VCODEC_MMU_INT_MASK

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR enable the read bus interrupt source when this bit is set to 1'b1 |
| 0 | RW | 0x0 | PAGE_FAULT enable the page fault interrupt source when this bit is set to 1'b1 |

VCODEC_MMU_INT_STATUS

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RO | 0x0 | READ_BUS_ERROR 1'b1:read bus error status |
| 0 | RO | 0x0 | PAGE_FAULT 1'b1:page fault |

VCODEC_MMU_AUTO_GATING

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | mmu_auto_clkgating when it is 1'b1, the mmu will auto gating it self |

4.5.6 VDPU121 Cache Register Summary

| Name | Offset | Size | Reset Value | Description |
|---------------------------------|--------|------|-------------|--|
| <u>pref_cache VERSION</u> | 0x0000 | W | 0xcac20101 | VERSION register |
| <u>pref_cache SIZE</u> | 0x0004 | W | 0x06110206 | L2 cache SIZE |
| <u>pref_cache STATUS</u> | 0x0008 | W | 0x00000000 | Status register |
| <u>pref_cache COMMAND</u> | 0x0010 | W | 0x00000000 | Command setting register |
| <u>pref_cache CLEAR PAGE</u> | 0x0014 | W | 0x00000000 | clear page register |
| <u>pref_cache MAX READS</u> | 0x0018 | W | 0x0000001c | maximum read register |
| <u>pref_cache PERFCNT_SR_C0</u> | 0x0020 | W | 0x00000000 | performance counter 0 source register |
| <u>pref_cache PERFCNT_VAL_0</u> | 0x0024 | W | 0x00000000 | performance counter 0 value register |
| <u>pref_cache PERFCNT_SR_C1</u> | 0x0028 | W | 0x00000000 | This register holds all the possible source values for Performance Counter 00: total clock cycles1: active clock cycles2: read transactions, master3: word reads, master4: read transactions, slave5: word reads, slave6: read hit, slave7: read misses, slave8: read invalidates, slave9: cacheable read transactions, slave10: bad hit number, slave |
| <u>pref_cache PERFCNT_VAL_1</u> | 0x002c | W | 0x00000000 | performance counter 1 value register |

Notes: **S**-ize: **B**- Byte (8 bits) access, **H****W**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

4.5.7 VDPU121 Cache Detail Register Description

pref_cache VERSION

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RO | 0xcac2 | PRODUCT_ID Field0000 Description |
| 15:8 | RO | 0x01 | VERSION_MAJOR Field0000 Description |
| 7:0 | RO | 0x01 | VERSION_MINOR Field0000 Description |

pref_cache SIZE

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x06 | External_bus_width Log2 external bus width in bits |
| 23:16 | RO | 0x11 | CACHE_SIZE Log2 cache size in bytes |
| 15:8 | RO | 0x02 | ASSOCIATIVITY Log2 associativity |
| 7:0 | RO | 0x06 | LINE_SIZE Log2 line size in bytes |

pref cache STATUS

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RO | 0x0 | DATA_BUSY set when the cache is busy handling data |
| 0 | RO | 0x0 | CMD_BUSY set when the cache is busy handling commands |

pref cache COMMAND

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | reserved |
| 5:4 | RW | 0x0 | sw_addrb_sel 2'b00: to sel b[14:6] 2'b01: to sel b[15:9], b[7:6] 2'b10: to sel b[16:10], b[7:6] 2'b11: to sel b[17:11], b[7:6] |
| 3 | RO | 0x0 | reserved |
| 2:0 | WO | 0x0 | COMMAND The possible command is 1 = Clear entire cache |

pref cache CLEAR PAGE

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | WO | 0x00000000 | CLEAR_PAGE writing an address, invalidates all lines in that page from the cache |

pref cache MAX READS

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x1c | MAX_READS Limit the number of outstanding read transactions to this amount |

pref cache PERFCNT_SRC0

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | PERFCNT_SRC0 This register holds all the possible source values for Performance Counter 0 0: total clock cycles 1: active clock cycles 2: read transactions, master 3: word reads, master 4: read transactions, slave 5: word reads, slave 6: read hit, slave 7: read misses, slave 8: read invalidates, slave 9: cacheable read transactions, slave 10: bad hit nmber, slave |

pref cache PERFCNT_VAL0

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | PERFCNT_VAL0 Performance counter 0 value |

pref cache PERFCNT_SRC1

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | <p>PERFCNT_SRC1</p> <p>This register holds all the possible source values for Performance Counter 1</p> <ul style="list-style-type: none"> 0: total clock cycles 1: active clock cycles 2: read transactions, master 3: word reads, master 4: read transactions, slave 5: word reads, slave 6: read hit, slave 7: read misses, slave 8: read invalidates, slave 9: cacheable read transactions, slave 10: bad hit nmber, slave |

pref cache PERFCNT VAL1

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>PERFCNT_VAL1</p> <p>Performance counter 1 value</p> |

4.5.8 VEPU121 Function Config Register Summary

| Name | Offset | Size | Reset Value | Description |
|--------------|--------|------|-------------|---|
| VEPU_swreg_0 | 0x0000 | W | 0x00000000 | 1st quantization for jpeg lumin table |
| VEPU_swreg_1 | 0x0004 | W | 0x00000000 | 2st quantization for jpeg lumin table |
| VEPU_swreg_2 | 0x0008 | W | 0x00000000 | 3st quantization for jpeg lumin table |
| VEPU_swreg_3 | 0x000c | W | 0x00000000 | 4st quantization for jpeg lumin table |
| VEPU_swreg_4 | 0x0010 | W | 0x00000000 | 5st quantization for jpeg lumin table |
| VEPU_swreg_5 | 0x0014 | W | 0x00000000 | 6st quantization for jpeg lumin table/part 1 for qp round |
| VEPU_swreg_6 | 0x0018 | W | 0x00000000 | 7st quantization for jpeg lumin table |
| VEPU_swreg_7 | 0x001c | W | 0x00000000 | 8st quantization for jpeg lumin table |
| VEPU_swreg_8 | 0x0020 | W | 0x00000000 | 9st quantization for jpeg lumin table |

| Name | Offset | Size | Reset Value | Description |
|---------------|--------|------|-------------|--|
| VEPU_swreg_9 | 0x0024 | W | 0x00000000 | 10st quantization for jpeg lumin table |
| VEPU_swreg_10 | 0x0028 | W | 0x00000000 | 11st quantization for jpeg lumin table |
| VEPU_swreg_11 | 0x002c | W | 0x00000000 | 12st quantization for jpeg lumin table |
| VEPU_swreg_12 | 0x0030 | W | 0x00000000 | 13st quantization for jpeg lumin table |
| VEPU_swreg_13 | 0x0034 | W | 0x00000000 | 14st quantization for jpeg lumin table |
| VEPU_swreg_14 | 0x0038 | W | 0x00000000 | 15st quantization for jpeg lumin table |
| VEPU_swreg_15 | 0x003c | W | 0x00000000 | 16st quantization for jpeg lumin table |
| VEPU_swreg_16 | 0x0040 | W | 0x00000000 | 1st quantization for jpeg chroma table |
| VEPU_swreg_17 | 0x0044 | W | 0x00000000 | 2st quantization for jpeg chroma table |
| VEPU_swreg_18 | 0x0048 | W | 0x00000000 | 3st quantization for jpeg chroma table |
| VEPU_swreg_19 | 0x004c | W | 0x00000000 | 4st quantization for jpeg chroma table |
| VEPU_swreg_20 | 0x0050 | W | 0x00000000 | 5st quantization for jpeg chroma table |
| VEPU_swreg_21 | 0x0054 | W | 0x00000000 | 6st quantization for jpeg chroma table |
| VEPU_swreg_22 | 0x0058 | W | 0x00000000 | 7st quantization for jpeg chroma table |
| VEPU_swreg_23 | 0x005c | W | 0x00000000 | 8st quantization for jpeg chroma table/part 3 for qp round |
| VEPU_swreg_24 | 0x0060 | W | 0x00000000 | 9st quantization for jpeg chroma table |
| VEPU_swreg_25 | 0x0064 | W | 0x00000000 | 10st quantization for jpeg chroma table |
| VEPU_swreg_26 | 0x0068 | W | 0x00000000 | 11st quantization for jpeg chroma table |
| VEPU_swreg_27 | 0x006c | W | 0x00000000 | 12st quantization for jpeg chroma |
| VEPU_swreg_28 | 0x0070 | W | 0x00000000 | 13st quantization for jpeg chroma |
| VEPU_swreg_29 | 0x0074 | W | 0x00000000 | 14st quantization for jpeg chroma |

| Name | Offset | Size | Reset Value | Description |
|---------------------|--------|------|-------------|--|
| VEPU_swreg_30 | 0x0078 | W | 0x00000000 | 15st quantization for jpeg chroma |
| VEPU_swreg_31 | 0x007c | W | 0x00000000 | 16st quantization for jpeg chroma |
| VEPU_swreg_44 | 0x00b0 | W | 0x00000000 | Intra slice bitmap |
| VEPU_swreg_45 | 0x00b4 | W | 0x00000000 | Intra slice bitmap1 |
| VEPU_swreg_46 | 0x00b8 | W | 0x00000000 | intra macro block sellect register |
| VEPU_swreg_47 | 0x00bc | W | 0x00000000 | CIR intra control register |
| VEPU_swreg_48 | 0x00c0 | W | 0x00000000 | base addr for input luma |
| VEPU_swreg_49 | 0x00c4 | W | 0x00000000 | base address for input cb |
| VEPU_swreg_50 | 0x00c8 | W | 0x00000000 | input cr start address |
| VEPU_swreg_51 | 0x00cc | W | 0x00000000 | stream header bits left register |
| VEPU_swreg_52 | 0x00d0 | W | 0x00000000 | stream header bits left register |
| VEPU_swreg_53 | 0x00d4 | W | 0x00000000 | stream buffer register |
| VEPU_swreg_54 | 0x00d8 | W | 0x01010000 | axi control register |
| VEPU_swreg_55 | 0x00dc | W | 0x00000000 | qp related |
| VEPU_swreg_56 | 0x00e0 | W | 0x00000000 | the luma reference frame start address |
| VEPU_swreg_57 | 0x00e4 | W | 0x00000000 | the chroma reference frame start address |
| VEPU_swreg_58 | 0x00e8 | W | 0x00000000 | the result of qp sum div2 |
| VEPU_swreg_59 | 0x00ec | W | 0x00000000 | h264 slice ctrl |
| VEPU_swreg_60 | 0x00f0 | W | 0x00000000 | spill ctrl |
| VEPU_swreg_61 | 0x00f4 | W | 0x00000000 | input luminance information |
| VEPU_swreg_62 | 0x00f8 | W | 0x00000000 | rlc_sum |
| VEPU_swreg_63 | 0x00fc | W | 0x00000000 | the reconstructed luma start address |
| VEPU_swreg_64 | 0x0100 | W | 0x00000000 | the reconstructed chroma start address |
| VEPU_swreg_65_reuse | 0x0104 | W | 0x00000000 | checkpoint 1 and 2 |
| VEPU_swreg_66_reuse | 0x0108 | W | 0x00000000 | checkpoint 3 and 4 |
| VEPU_swreg_67_reuse | 0x010c | W | 0x00000000 | checkpoint 5 and 6 |
| VEPU_swreg_68_reuse | 0x0110 | W | 0x00000000 | checkpoint 7 and 8 |
| VEPU_swreg_69_reuse | 0x0114 | W | 0x00000000 | checkpoint 9 and 10 |
| VEPU_swreg_70_reuse | 0x0118 | W | 0x00000000 | checkpoint word error 1 and 2 |
| VEPU_swreg_71_reuse | 0x011c | W | 0x00000000 | checkpoint word error 1 and 2 |
| VEPU_swreg_72_reuse | 0x0120 | W | 0x00000000 | checkpoint word error 1 and 2 |
| VEPU_swreg_73_reuse | 0x0124 | W | 0x00000000 | checkpoint delta QP register |
| VEPU_swreg_74 | 0x0128 | W | 0x00000000 | input image format |
| VEPU_swreg_75 | 0x012c | W | 0x00000000 | intra/inter mode |
| VEPU_swreg_76_reuse | 0x0130 | W | 0x00000000 | encoder control regisiter 0 |
| VEPU_swreg_77 | 0x0134 | W | 0x00000000 | output stream start address |
| VEPU_swreg_78 | 0x0138 | W | 0x00000000 | output control start address |

| Name | Offset | Size | Reset Value | Description |
|----------------------|--------|------|-------------|--|
| VEPU_swreg_79 | 0x013c | W | 0x00000000 | next picture luminance start address |
| VEPU_swreg_80 | 0x0140 | W | 0x00000000 | Base address for MV output |
| VEPU_swreg_81 | 0x0144 | W | 0x00000000 | the cabac table start address |
| VEPU_swreg_82 | 0x0148 | W | 0x00000000 | ROI area register |
| VEPU_swreg_83 | 0x014c | W | 0x00000000 | the second of ROI area register |
| VEPU_swreg_84 | 0x0150 | W | 0x00000000 | Stabilization matrix1 |
| VEPU_swreg_85 | 0x0154 | W | 0x00000000 | Stabilization matrix2 |
| VEPU_swreg_86 | 0x0158 | W | 0x00000000 | Stabilization matrix3 |
| VEPU_swreg_87 | 0x015c | W | 0x00000000 | Stabilization matrix4 |
| VEPU_swreg_88 | 0x0160 | W | 0x00000000 | Stabilization matrix5 |
| VEPU_swreg_89 | 0x0164 | W | 0x00000000 | Stabilization matrix6 |
| VEPU_swreg_90 | 0x0168 | W | 0x00000000 | Stabilization matrix7 |
| VEPU_swreg_91 | 0x016c | W | 0x00000000 | Stabilization matrix8 |
| VEPU_swreg_92 | 0x0170 | W | 0x00000000 | Stabilization matrix9 |
| VEPU_swreg_93 | 0x0174 | W | 0x00000000 | the output of Stabilization motion sum |
| VEPU_swreg_94 | 0x0178 | W | 0x00000000 | output of Stabilization |
| VEPU_swreg_95 | 0x017c | W | 0x00000000 | RGB to YUV conversion coefficient register |
| VEPU_swreg_96 | 0x0180 | W | 0x00000000 | RGB to YUV conversion coefficient register |
| VEPU_swreg_97 | 0x0184 | W | 0x00000000 | RGB to YUV conversion coefficient register |
| VEPU_swreg_98 | 0x0188 | W | 0x00000000 | RGA MASK |
| VEPU_swreg_99 | 0x018c | W | 0x00000000 | mv related |
| VEPU_swreg_100_reuse | 0x0190 | W | 0x00000000 | QP register |
| VEPU_swreg_101_read | 0x0194 | W | 0x1f522780 | hw config reg |
| VEPU_swreg_102 | 0x0198 | W | 0x00000000 | mvc related |
| VEPU_swreg_103 | 0x019c | W | 0x00000000 | encoder start |
| VEPU_swreg_104 | 0x01a0 | W | 0x00000000 | mb control register |
| VEPU_swreg_105 | 0x01a4 | W | 0x00000000 | swap ctrl register |
| VEPU_swreg_106_reuse | 0x01a8 | W | 0x00000000 | encoder control register 1 |
| VEPU_swreg_107_reuse | 0x01ac | W | 0x00000000 | JPEG control register |
| VEPU_swreg_108_reuse | 0x01b0 | W | 0x00000000 | intra slice bmp2 |
| VEPU_swreg_109 | 0x01b4 | W | 0x00001000 | encoder status |
| VEPU_swreg_110_read | 0x01b8 | W | 0x48311220 | product ID |

| Name | Offset | Size | Reset Value | Description |
|--------------------|--------|------|-------------|--|
| VEPU_swreg_120_183 | 0x01e0 | W | 0x00000000 | addr range : 0x01e0~0x02dcswreg120: DMV 4p/1p penalty table valuesswreg121: DMV 4p/1p penalty table valuesswreg122: DMV 4p/1p penalty table valuesswreg123: DMV 4p/1p penalty table values.....swreg183: DMV 4p/1p penalty table values |

Notes: **Size:** **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

4.5.9 VEPU121 Function Config Detail Register Description

VEPU_swreg_0

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant1 jpeg luma quantization 1 |

VEPU_swreg_1

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant2 jpeg luma quantization 2 |

VEPU_swreg_2

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant3 jpeg luma quantization 3 |

VEPU_swreg_3

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant4 jpeg luma quantization 4 |

VEPU_swreg_4

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant5 jpeg luma quantization 5 |

VEPU_swreg_5

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant6 jpeg luma quantization 6 |

VEPU_swreg_6

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant7 jpeg luma quantization 7 |

VEPU_swreg_7

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant8 jpeg luma quantization 8 |

VEPU_swreg_8

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant9 jpeg luma quantization 9 |

VEPU_swreg_9

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant10 jpeg luma quantization 10 |

VEPU_swreg_10

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant11 jpeg luma quantization 11 |

VEPU_swreg_11

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant12 jpeg luma quantization 12 |

VEPU_swreg_12

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant13 jpeg luma quantization 13 |

VEPU_swreg_13

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant14 jpeg luma quantization 14 |

VEPU_swreg_14

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_luma_quant15 jpeg luma quantization 15 |

VEPU_swreg_15

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | sw_jpeg_luma_quant16 jpeg luma quantization 16 |

VEPU_swreg_16

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant1 jpeg chroma quantization 1 |

VEPU_swreg_17

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant2 jpeg chroma quantization 2 |

VEPU_swreg_18

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant3 jpeg chroma quantization 3 |

VEPU_swreg_19

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant4 jpeg chroma quantization 4 |

VEPU_swreg_20

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant5 jpeg chroma quantization 5 |

VEPU_swreg_21

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant6 jpeg chroma quantization 6 |

VEPU_swreg_22

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant7 jpeg chroma quantization 7 |

VEPU_swreg_23

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant8 jpeg chroma quantization 8 |

VEPU_swreg_24

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant9 jpeg chroma quantization 9 |

VEPU_swreg_25

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant10 jpeg chroma quantization 10 |

VEPU_swreg_26

Address: Operational Base + offset (0x0068)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:9 | RO | 0x0 | reserved |
| 8:0 | RW | 0x000 | sw_jpeg_chroma_quant11 jpeg chroma quantization 11 |

VEPU_swreg_27

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant11 jpeg chroma quantization 11 |

VEPU_swreg_28

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x000 | sw_jpeg_chroma_quant13 jpeg chroma quantization 13 |

VEPU_swreg_29

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant14 jpeg chroma quantization 14 |

VEPU_swreg_30

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x000 | sw_jpeg_chroma_quant15 jpeg chroma quantization 15 |

VEPU_swreg_31

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | sw_jpeg_chroma_quant16 jpeg chroma quantization 16 |

VEPU_swreg_44

Address: Operational Base + offset (0x00b0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | intra_slice bmp0 bit0 : slices0 bit1 : slices1 bit2 : slices2 bit31 : slices31 |

VEPU_swreg_45

Address: Operational Base + offset (0x00b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | intra_slice_bmp1 bit0 : slices32 bit1 : slices33 bit2 : slices34 bit31 : slices63 |

VEPU_swreg_46

Address: Operational Base + offset (0x00b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x00 | intra_up_mb_area The top intra macro block's area used in row |
| 23:16 | RW | 0x00 | intra_down_mb_area The bottom intra macro block's area used in row |
| 15:8 | RW | 0x00 | intra_left_mb_area The left intra macro block's area used in column |
| 7:0 | RW | 0x00 | intra_right_mb_area The right intra macro block's area used in column |

VEPU_swreg_47

Address: Operational Base + offset (0x00bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | cir_first_intra 0:disable other:enable and be set |
| 15:0 | RW | 0x0000 | cir_intra_mb_itvl 0:disable other: enable and be set |

VEPU_swreg_48

Address: Operational Base + offset (0x00c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | luma_in_st_adr input luma start address |

VEPU_swreg_49

Address: Operational Base + offset (0x00c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | cb_in_st_adr input cb start address |

VEPU_swreg_50

Address: Operational Base + offset (0x00c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | cr_in_st_addr input cr start address |

VEPU_swreg_51

Address: Operational Base + offset (0x00cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | strm_header_left_hbits the high 32 bit of stram header be left |

VEPU_swreg_52

Address: Operational Base + offset (0x00d0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | strm_header_left_lbits the low 32 bit of stram header be left |

VEPU_swreg_53

Address: Operational Base + offset (0x00d4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | strm_bufsize_lmt the limit size of steam buffer |

VEPU_swreg_54

Address: Operational Base + offset (0x00d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x01 | axi_rd_id if config 0,it will be modify as 1 by HW auto |
| 23:16 | RW | 0x01 | axi_wr_id if config 0,it will be modify as 1 by HW auto |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | burst_len burst length |
| 7:3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | burst_incr_mod_sel 0: single burst selected 1: incr burst selected |
| 1 | RW | 0x0 | burst_discard 0:disable ,off 1:enable,on |
| 0 | RW | 0x0 | burst_disable 0: enable 1:disable |

VEPU_swreg_55

Address: Operational Base + offset (0x00dc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | reserved |
| 15:12 | RW | 0x0 | roi_dlt_qp1 1st for delta qp for roi |
| 11:8 | RW | 0x0 | roi_dlt_qp2 2st for delta qp for roi |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | qp_adjst signed register; range from -8 to 7 |

VEPU_swreg_56

Address: Operational Base + offset (0x00e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | luma_ref_st_adr the luma reference frame start address |

VEPU_swreg_57

Address: Operational Base + offset (0x00e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | chroma_ref_st_adr the chroma reference frame start address |

VEPU_swreg_58

Address: Operational Base + offset (0x00e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:11 | RW | 0x0000000 | qp_sum_div2 the result of (qp sum)/2 |
| 10:0 | RO | 0x0 | reserved |

VEPU_swreg_59

Address: Operational Base + offset (0x00ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28 | RW | 0x0 | h264_qurt_pixmv_dis 0:default,enable 1:disable |
| 27:26 | RO | 0x0 | reserved |
| 25:24 | RW | 0x0 | dblking_flt_mode 0 : enabled 1 : disabled 2 : disabled on slice |
| 23 | RO | 0x0 | reserved |
| 22:21 | RW | 0x0 | h264_cabac_idc 0,1,2: used 3: no use |
| 20 | RW | 0x0 | entry_code_fmt h.264: 0: cavlc 1: cabac |
| 19:18 | RO | 0x0 | reserved |
| 17 | RW | 0x0 | h264_trfmod_8x8 on-off for 8x8 transform used in h264 |
| 16 | RW | 0x0 | h264_res_intermod_4x4 the restriction inter mode selected in 4x4 block |
| 15 | RW | 0x0 | h264_strm_mod_sel 0 : NAL unit ; 1 : BYTE |
| 14:8 | RW | 0x00 | h264_slice_num 0=one slice in current picture 1=two slice in current picture |
| 7:0 | RO | 0x0 | reserved |

VEPU_swreg_60

Address: Operational Base + offset (0x00f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x00 | strm_st_offset |
| 15:8 | RW | 0x00 | skip_mb_mode H.264:SKIP macroblock mode |
| 7:6 | RO | 0x0 | reserved |
| 5:4 | RW | 0x0 | right_spill div4 value range:0~3 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 3:0 | RW | 0x0 | bot_spill the bottom edge of image for spill pixels |

VEPU_swreg_61

Address: Operational Base + offset (0x00f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------|
| 31:23 | RO | 0x0 | reserved |
| 22:20 | RW | 0x0 | offset_in_chroma byte unit |
| 19 | RO | 0x0 | reserved |
| 18:16 | RW | 0x0 | offset_in_luma byte unit |
| 15:14 | RO | 0x0 | reserved |
| 13:0 | RW | 0x0000 | row_len_in_luma |

VEPU_swreg_62

Address: Operational Base + offset (0x00f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:22 | RO | 0x0 | reserved |
| 21:0 | RW | 0x0000000 | rlc_sum rlc_sum |

VEPU_swreg_63

Address: Operational Base + offset (0x00fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | recon_luma_st_adr the reconstructed luma start address |

VEPU_swreg_64

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | recon_chroma_st_adr the reconstructed chroma start address |

VEPU_swreg_65_reuse

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_chkpt_1 1st word used for check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_chkpt_2 2st word used for check point used in h.264 |

VEPU_swreg_66_reuse

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_chkpt_3 3st word used for check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_chkpt_4 4st word used for check point used in h.264 |

VEPU_swreg_67_reuse

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_chkpt_5 5st word used for check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_chkpt_6 6st word used for check point used in h.264 |

VEPU_swreg_68_reuse

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_chkpt_7 7st word used for check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_chkpt_8 8st word used for check point used in h.264 |

VEPU_swreg_69_reuse

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_chkpt_9 9st word used for check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_chkpt_10 10st word used for check point used in h.264 |

VEPU_swreg_70_reuse

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_errchkpt_1 1st word error check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_errchkpt_2 2st word error check point used in h.264 |

VEPU_swreg_71_reuse

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_errchkpt_3 3st word error check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_errchkpt_4 4st word error check point used in h.264 |

VEPU_swreg_72_reuse

Address: Operational Base + offset (0x0120)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | h264_errchkpt_5 5st word error check point used in h.264 |
| 15:0 | RW | 0x0000 | h264_errchkpt_6 6st word error check point used in h.264 |

VEPU_swreg_73_reuse

Address: Operational Base + offset (0x0124)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:24 | RW | 0x0 | chkqp_1 1st for delta qp check point |
| 23:20 | RW | 0x0 | chkqp_2 2st for delta qp check point |
| 19:16 | RW | 0x0 | chkqp_3 3st for delta qp check point |
| 15:12 | RW | 0x0 | chkqp_4 4st for delta qp check point |
| 11:8 | RW | 0x0 | chkqp_5 5st for delta qp check point |
| 7:4 | RW | 0x0 | chkqp_6 6st for delta qp check point |
| 3:0 | RW | 0x0 | chkqp_7 7st for delta qp check point |

VEPU_swreg_74

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:24 | RW | 0x00 | mad_thsld value = (MAD threshold)/256 |
| 23:16 | RW | 0x00 | encoderd_slices the number of encoder slices which used in h.264 |
| 15:8 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 7:4 | RW | 0x0 | img_fmt_in YUV420P YUV420SP YUV422 UYVY422 RGB565 RGB444 RGB888 RGB101010 |
| 3:2 | RW | 0x0 | img_in_rot 0 : no rotation 1 : rotate right 90 degress 2 : rotate left 90 degress |
| 1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | nal_mode the output of NAL size to base control |

VEPU_swreg_75

Address: Operational Base + offset (0x012c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RW | 0x0000 | intramod_16x16 |
| 15:0 | RW | 0x0000 | intermod the intra/inter selection for inter macro block mode favor |

VEPU_swreg_76_reuse

Address: Operational Base + offset (0x0130)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:26 | RW | 0x00 | pps_init_qp pps init qp in picture used in h264 range : 0~51 |
| 25:22 | RW | 0x0 | slice_flt_alpha offset div2 range : -6~6 |
| 21:18 | RW | 0x0 | slice_flt_beta config value = (real value)/2 signed register range : -6 ~6 |
| 17:13 | RW | 0x00 | qp_offset_ch signed register range : -12~12 |
| 12:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | sw_qpass |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 7:5 | RO | 0x0 | reserved |
| 4:1 | RW | 0x0 | idr_picid IDR pic ID |
| 0 | RW | 0x0 | constr_intra_pred constrained intra prediction |

VEPU_swreg_77

Address: Operational Base + offset (0x0134)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | output_strm_st_adr output stream start address |

VEPU_swreg_78

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | output_ctrl_st_adr output control start address |

VEPU_swreg_79

Address: Operational Base + offset (0x013c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | next_luma_st_adr next picture luminance start address |

VEPU_swreg_80

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:0 | RW | 0x00000000 | mv_out_st_adr |

VEPU_swreg_81

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | cabac_table_st_adr H264: cabac table |

VEPU_swreg_82

Address: Operational Base + offset (0x0148)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------|
| 31:24 | RW | 0x00 | first_roi_tmb (inside area) |
| 23:16 | RW | 0x00 | first_roi_bmb (outside area) |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 15:8 | RW | 0x00 | first_roi_lmb qp=qp + roi1_Delta_Qp (inside area) |
| 7:0 | RW | 0x00 | first_roi_rmb qp=qp - roi1_Delta_Qp (outside area) |

VEPU_swreg_83

Address: Operational Base + offset (0x014c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0x00 | second_roi_rmb (inside area) |
| 23:16 | RW | 0x00 | second_roi_bmb (outside area) |
| 15:8 | RW | 0x00 | second_roi_lmb qp=qp + roi1_Delta_Qp (inside area) |
| 7:0 | RW | 0x00 | second_roi_tmb qp=qp - roi1_Delta_Qp (outside area) |

VEPU_swreg_84

Address: Operational Base + offset (0x0150)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix1 (position@ up-left) |

VEPU_swreg_85

Address: Operational Base + offset (0x0154)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix2 (position @ up) |

VEPU_swreg_86

Address: Operational Base + offset (0x0158)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix3 (position @up-right) |

VEPU_swreg_87

Address: Operational Base + offset (0x015c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix4 (position @ left) |

VEPU_swreg_88

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix5 (position @GMV) |

VEPU_swreg_89

Address: Operational Base + offset (0x0164)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix6 (position@right) |

VEPU_swreg_90

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix7 (position@down-left) |

VEPU_swreg_91

Address: Operational Base + offset (0x016c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix8 (position@down) |

VEPU_swreg_92

Address: Operational Base + offset (0x0170)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:26 | RW | 0x00 | stab_gmv_vrtl signed register range : -16~16 |
| 25:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x000000 | stab_matrix9 (position@down-right) |

VEPU_swreg_93

Address: Operational Base + offset (0x0174)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | stab_motion_sum read value = (real value)/8 range : 0~1089*253*255*53/8 |

VEPU_swreg_94

Address: Operational Base + offset (0x0178)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RW | 0x0000000 | stab_min_value range : 0~255*253*253 |
| 7:6 | RW | 0x0 | stab_mod_sel 0 : disabled 1 : stab only 2 : stab+encode |
| 5:0 | RW | 0x00 | stab_hor_gmv signed register range : -16~16 |

VEPU_swreg_95

Address: Operational Base + offset (0x017c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RW | 0x0000 | rgb2yuv_coe2 the 2st conversion coefficien for RGB to YUV |
| 15:0 | RW | 0x0000 | rgb2yuv_coe1 the 1st conversion coefficien for RGB to YUV |

VEPU_swreg_96

Address: Operational Base + offset (0x0180)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RW | 0x0000 | rgb2yuv_coe5 the 5st conversion coefficien for RGB to YUV |
| 15:0 | RW | 0x0000 | rgb2yuv_coe3 the 3st conversion coefficien for RGB to YUV |

VEPU_swreg_97

Address: Operational Base + offset (0x0184)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RO | 0x0 | reserved |
| 15:0 | RW | 0x0000 | rgb2yuv_coe6 the 6st conversion coefficien for RGB to YUV |

VEPU_swreg_98

Address: Operational Base + offset (0x0188)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | bcmpt_mask_position range : 0~31 |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x00 | gcmpt_mask_position range : 0~31 |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | rcmpt_mask_position range : 0~31 |

VEPU_swreg_99

Address: Operational Base + offset (0x018c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30:21 | RW | 0x000 | mv_1p_ply differential MV penalty for 1p |
| 20:11 | RW | 0x000 | mv_1p_4p_ply ME. DMVPenaltyQp |
| 10:1 | RW | 0x000 | mv_4p_ply |
| 0 | RW | 0x0 | mutimv_en on-off flag for using exceed one mv every mb |

VEPU_swreg_100_reuse

Address: Operational Base + offset (0x0190)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:26 | RW | 0x00 | h264_init_luma_qp range: 0~51 |
| 25:20 | RW | 0x00 | h264_max_qp range : 0~51 |
| 19:14 | RW | 0x00 | h264_min_qp range:0~51 |
| 13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | h264_chkpt_distance checkpoint distance for macro block |

VEPU_swreg_101_read

Address: Operational Base + offset (0x0194)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x1f522 | HW_CONFIG Field0000 Description |
| 11:0 | RO | 0x780 | MAX_VID_WIDTH Field0000 Description |

VEPU_swreg_102

Address: Operational Base + offset (0x0198)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:20 | RW | 0x0 | mv_favor_16x16 value = (real value)/2. |
| 19:11 | RW | 0x000 | mv_ply_4x4 |
| 10:8 | RW | 0x0 | mvc_view_id |
| 7 | RW | 0x0 | mvc_anchor_pic_flag |
| 6:4 | RW | 0x0 | mvc_priority_id |
| 3:1 | RW | 0x0 | mvc_temporal_id |
| 0 | RW | 0x0 | mvc_inter_view_flag MVC inter_view_flag. |

VEPU_swreg_103

Address: Operational Base + offset (0x019c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:29 | RO | 0x0 | reserved |
| 28:20 | RW | 0x000 | enc_height lum height (macroblock unit) H264: [6..255] JPEG: [6..511] |
| 19:17 | RO | 0x0 | reserved |
| 16:8 | RW | 0x000 | enc_width lum width (macroblock unit) H264: range : 9~255 JPEG: range : 6~511 |
| 7:6 | RW | 0x0 | enc_frame_type 0: INTER 1: INTRA(IDR) 2: MVC-INTER |
| 5:4 | RW | 0x0 | enc_fmt 2 : JPEG 3 : H264 |
| 3:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | enc_en encoder enable |

VEPU swreg 104

Address: Operational Base + offset (0x01a0)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|------------------------------|
| 31:16 | RW | 0x0000 | mb_count_out mb_count_out |
| 15:0 | RW | 0x0000 | mb_cnt macroblock_count |

VEPU swreg 105

Address: Operational Base + offset (0x01a4)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31 | RW | 0x0 | swap8_in 0: no swap 1: swap 8bit |
| 30 | RW | 0x0 | swap16_in 0: no swap 1: swap 16bit |
| 29 | RW | 0x0 | swap32_in 0: no swap 1: swap 32bit |
| 28 | RW | 0x0 | swap8_out 0: no swap 1: swap 8bit |
| 27 | RW | 0x0 | swap16_out 0: no swap 1: swap 16bit |
| 26 | RW | 0x0 | swap32_out 0: no swap 1: swap 32bit |
| 25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | test_irq test irq |
| 23:20 | RW | 0x0 | test_counter test counter |
| 19 | RW | 0x0 | coher_test_reg test register coherency |
| 18 | RW | 0x0 | coher_test_mem test memory coherency |
| 17:0 | RW | 0x00000 | test_len test data length |

VEPU swreg 106 reuse

Address: Operational Base + offset (0x01a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0x00 | pic_para_id |
| 23:16 | RW | 0x00 | intra_pred_mode H.264 intra prediction previous 4x4 mode favor |
| 15:0 | RW | 0x0000 | frame_num H.264 frame number |

VEPU_swreg_107_reuse

Address: Operational Base + offset (0x01ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:20 | RW | 0x000 | mv_ply_16x8_8x16 Penalty for using 16x8 or 8x16 MV |
| 19:10 | RW | 0x000 | mv_ply_8x8 Penalty for using 8x8 MV |
| 9:0 | RW | 0x000 | mv_ply_8x4_4x8 Penalty for using 8x4 or 4x8 MV. |

VEPU_swreg_108_reuse

Address: Operational Base + offset (0x01b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | intra_slice_bmp2 bit0 : slices64 bit1 : slices65 bit2 : slices66 bit31 : slices95 |

VEPU_swreg_109

Address: Operational Base + offset (0x01b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28 | RW | 0x0 | int_non Field0000 Description |
| 27:25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | mv_sad_wren |
| 23:21 | RO | 0x0 | reserved |
| 20 | RW | 0x0 | rocon_write_dis |
| 19:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | slice_rdyint_en enable slice ready interrupt |
| 15:13 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12 | RW | 0x1 | clk_gating_en default clk_gating_en = 1'b1 |
| 11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | int_timeout_en enable interrupt for timeout |
| 9 | RW | 0x0 | irq_clr |
| 8 | RW | 0x0 | irq_dis |
| 7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | irq_timeout |
| 5 | RW | 0x0 | irq_buffer_full |
| 4 | RW | 0x0 | irq_bus_error |
| 3 | RW | 0x0 | fuse_int Field0000 Description |
| 2 | RW | 0x0 | irq_slice_ready |
| 1 | RW | 0x0 | irq_frame_rdy |
| 0 | RW | 0x0 | enc_irq |

VEPU swreg 110 read

Address: Operational Base + offset (0x01b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------|
| 31:16 | RO | 0x4831 | prod_id Product ID |
| 15:12 | RO | 0x1 | major_num Major number |
| 11:4 | RO | 0x22 | minor_num Minor number |
| 3:0 | RO | 0x0 | synthesis |

VEPU swreg 120 183

Address: Operational Base + offset (0x01e0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | WO | 0x00000000 | dmv_ply_table addr range : 0x01e0~0x02dc swreg120: DMV 4p/1p penalty table values swreg121: DMV 4p/1p penalty table values swreg122: DMV 4p/1p penalty table values swreg123: DMV 4p/1p penalty table values swreg183: DMV 4p/1p penalty table values |

4.5.10 VEPU121 MMU Register Summary

| Name | Offset | Size | Reset Value | Description |
|-----------------------------------|--------|------|-------------|---|
| <u>VCODEC_MMU_DTE_ADDR</u> | 0x0000 | W | 0x00000000 | MMU current page Table addressIt is only can be written when MMU state is disable or page fault or mmu enable stall state |
| <u>VCODEC_MMU_STATUS</u> | 0x0004 | W | 0x00000018 | MMU status register |
| <u>VCODEC_MMU_COMMAND</u> | 0x0008 | W | 0x00000000 | MMU command register |
| <u>VCODEC_MMU_PAGE_FAULT_ADDR</u> | 0x000c | W | 0x00000000 | MMU logical address of last page fault |
| <u>VCODEC_MMU_ZAP_ONE_LINE</u> | 0x0010 | W | 0x00000000 | MMU Zap cache line register |
| <u>VCODEC_MMU_INT_RAW_STAT</u> | 0x0014 | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_INT_CLEA_R</u> | 0x0018 | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_INT_MASK</u> | 0x001c | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_INT_STAT_US</u> | 0x0020 | W | 0x00000000 | MMU raw interrupt status register |
| <u>VCODEC_MMU_AUTO_GATING</u> | 0x0024 | W | 0x00000001 | mmu auto gating |

Notes:Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

4.5.11 VEPU121 MMU Detail Register Description

VCODEC_MMU_DTE_ADDR

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | MMU_DTE_ADDR MMU current page Table address |

VCODEC_MMU_STATUS

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:11 | RO | 0x0 | reserved |
| 10:6 | RO | 0x00 | PAGE_FAULT_BUS_ID Index of master responsible for last page fault |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 5 | RO | 0x0 | PAGE_FAULT_IS_WRITE The direction of access for last page fault: 0 = Read 1 = Write |
| 4 | RO | 0x1 | REPLAY_BUFFER_EMPTY 1'b1: The MMU replay buffer is empty |
| 3 | RO | 0x1 | MMU_IDLE The MMU is idle when accesses are being translated and there are no unfinished translated accesses. 1'b1: MMU is idle |
| 2 | RO | 0x0 | STAIL_ACTIVE MMU stall mode currently enabled. The mode is enabled by command 1'b1: MMU is in stall active status |
| 1 | RO | 0x0 | PAGE_FAULT_ACTIVE MMU page fault mode currently enabled . The mode is enabled by command. 1'b1: page fault is active |
| 0 | RO | 0x0 | PAGING_ENABLED 1'b0: paging is disabled 1'b1: Paging is enabled |

VCODEC_MMU_COMMAND

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:3 | RO | 0x0 | reserved |
| 2:0 | WO | 0x0 | MMU_CMD MMU_CMD. This can be: 0: MMU_ENABLE_PAGING 1: MMU_DISABLE_PAGING 2: MMU_ENABLE_STALL 3: MMU_DISABLE_STALL 4: MMU_ZAP_CACHE 5: MMU_PAGE_FAULT_DONE 6: MMU_FORCE_RESET |

VCODEC_MMU_PAGE_FAULT_ADDR

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | PAGE_FAULT_ADDR address of last page fault |

VCODEC_MMU_ZAP_ONE_LINE

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | WO | 0x00000000 | MMU_ZAP_ONE_LINE address to be invalidated from the page table cache |

VCODEC_MMU_INT_RAWSTAT

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR read bus error status |
| 0 | RW | 0x0 | PAGE_FAULT page fault status |

VCODEC_MMU_INT_CLEAR

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | WO | 0x0 | READ_BUS_ERROR write 1 to clear read bus error |
| 0 | WO | 0x0 | PAGE_FAULT write 1 to page fault clear |

VCODEC_MMU_INT_MASK

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR enable the read bus interrupt source when this bit is set to 1'b1 |
| 0 | RW | 0x0 | PAGE_FAULT enable the page fault interrupt source when this bit is set to 1'b1 |

VCODEC_MMU_INT_STATUS

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1 | RO | 0x0 | READ_BUS_ERROR 1'b1:read bus error status |
| 0 | RO | 0x0 | PAGE_FAULT 1'b1:page fault |

VCODEC_MMU_AUTO_GATING

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | mmu_auto_clkgating when it is 1'b1, the mmu will auto gating it self |

4.6 Application Notes

4.6.1 H264/jpeg decoder or encoder Configuration flow

1. Prepare the decoder data in the DDR memory, and in decoder other than JPEG decoder, the input stream buffer should at least contain a slice or a frame data, otherwise the decoder will produce an interrupt and show error and then reset itself.

2. Config all the registers will be used. and please notice that which be list as follows:

In encoder---- We can configure the registers to control the input picture data format (such as endian and swap), but some input data format are fixed, such as cabac_table data. And the register VEPU_SWREG0~31 are JPEG quantization registers. They are write only registers. When you want to write these registers, you should first set VEPU_SWREG103[0] to 1'b0 and VEPU_SWREG103[5:4] to 2'b10(select JPEG mode).

In decoder---- The decoder can support ref buffer mode or cacheable mode, but they can't be both enabled. We can config the swreg57[28],swreg57[29] to enable cache and config the swreg65 to control the ref buffer.

4. You should config VDPU_SWREG57[0] as 1'b1 to enable video decoder. And config VDPU_SWREG41[0] as 1'b1 to enable pp. If pp performed in pipeline with decoder, you should config VDPU_SWREG41[4] as 1'b1 and then config VDPU_SWREG57[0] as 1'b1 to enable decoder and pp. VEPU_SWREG103[0] set to 1'b1 to enable encoder.

5. Wait for the frame interrupt, and then check if the frame decoder ready interrupt is right or not, after that, you can get the processed results in the target DDR

6. Clear all the interrupts, repeat step 2~5 to start a new frame decoder or encoder.

Chapter 5 Video Output Processor (VOP_LITE)

5.1 Overview

Video Output Processor is a video process engine and a display interface from memory frame buffer to display device. VOP is connected to an AHB bus through an AHB slave and AXI bus through an AXI master. The register setting is configured through the AHB slave interface and the display frame data is read through the AXI master interface.

5.1.1 Features

VOP_LITE supports the following features:

- Display interface
 - Parallel RGB LCD Interface: 18-bit(RGB666), 16-bit(RGB565)
 - MIPI interface
 - MCU interface
 - Max output resolution
 - ◆ 1920x1080 for MIPI
 - ◆ 1280x800 for RGB
- Display process
 - Background layer
 - ◆ programmable 24-bit color
 - Win1 layer
 - ◆ RGB888, ARGB888, RGB565
 - ◆ Support virtual display
 - ◆ 256 level alpha blending (pre-multiplied alpha support)
 - ◆ Transparency color key
 - ◆ RGB2YCbCr(BT601/BT709)
- Others
 - Support RGB or YUV domain overlay
 - BCSH(Brightness,Contrast,Saturation,Hue adjustment)
 - BCSH:YCbCr2RGB(rec601-mpeg/ rec601-jpeg/rec709)
 - BCSH:RGB2YCbCr(BT601/BT709)
 - Support Gamma adjust for PAD
 - Support dither down allegro RGB888to666 RGB888to565 & dither down frc (configurable) RGB888to666
 - Blank and black display
 - Standby mode
 - Support QoS request for higher bus priority for win1
 - Support DMA stop mode
 - Support all layers reg_done separately

5.2 Block Diagram

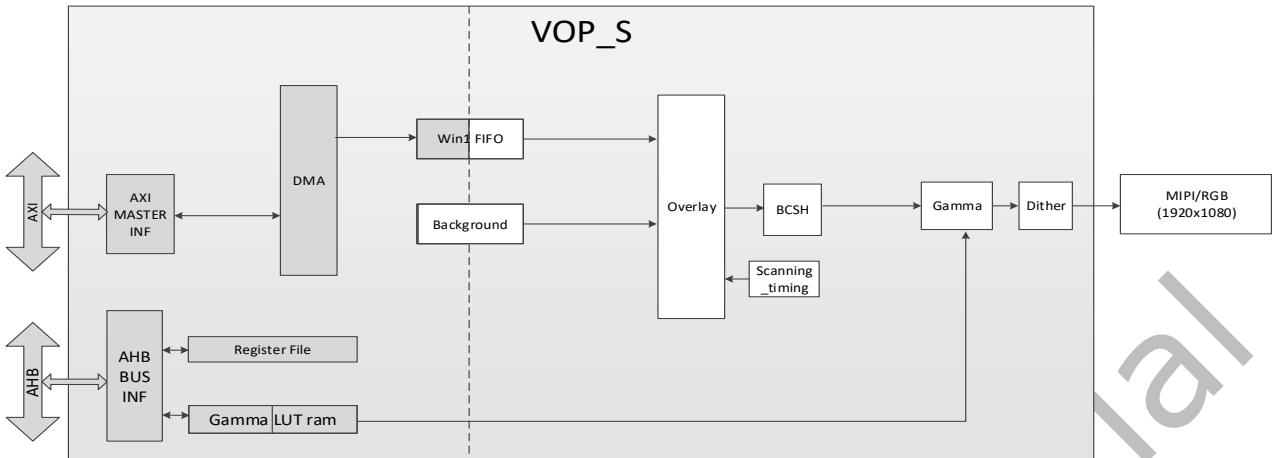


Fig. 5-1 VOP Block Diagram

5.3 Function Description

5.3.1 Data Format

VOP master read the frame data from the frame buffer in the system memory. There are total 3 formats supported in three layers.

- Win1: RGB888, ARGB888, RGB565

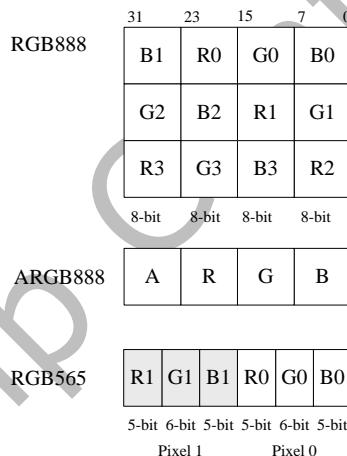


Fig. 5-2 VOP Frame Buffer Data Format

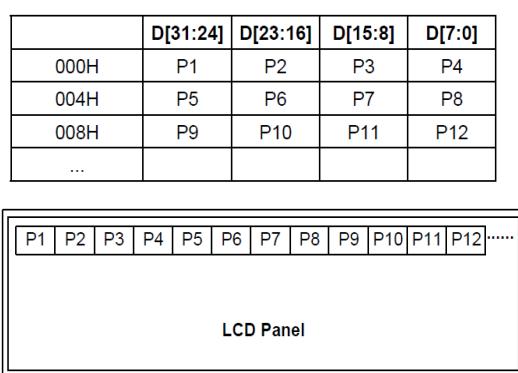


Fig. 5-3 VOP Win1 Palette (8bpp)

Data SWAP function

There are several swap options for different frame data formats. All the data swap types are in the following table.

Table 5-1 VOP Data Swap of Win1

| Data-swap | RB swap | Alpha swap | Y-M8 swap | CbCr swap |
|-----------|---------|------------|-----------|-----------|
| Win1 | yes | yes | No | No |

5.3.2 Data path

1. Internal DMA

Internal DMA can fetch the pixel data through AXI bus from system memory (DDR) for all the display layers. Data fetching is driven by display output requirement.

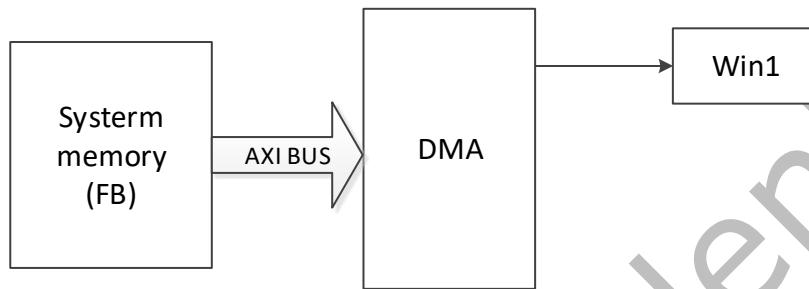


Fig. 5-4 VOP Internal DMA

5.3.3 Virtual display

Virtual display is supported in Win1. The active image is part of the virtual (original) image in frame buffer memory. The virtual width is indicated by setting WIN1_VIR_STRIDE for different data format.

The virtual stride should be multiples of word (32-bit). That means dummy bytes in the end of virtual line if the real pixels are not 32-bit aligned.

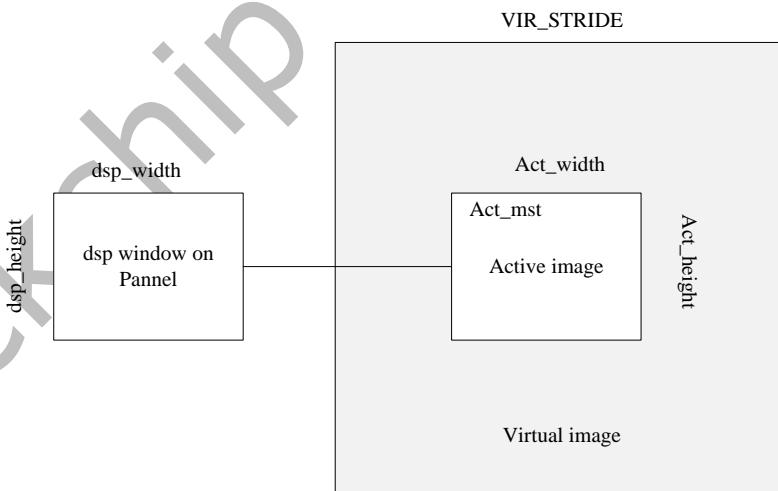


Fig. 5-5 VOP Virtual Display Mode

5.3.4 Overlay

1. Overlay display

There are totally 3 layers for overlay display: Background, Win1.

The background is a programmable solid color layer, which is always the bottom of the display screen.

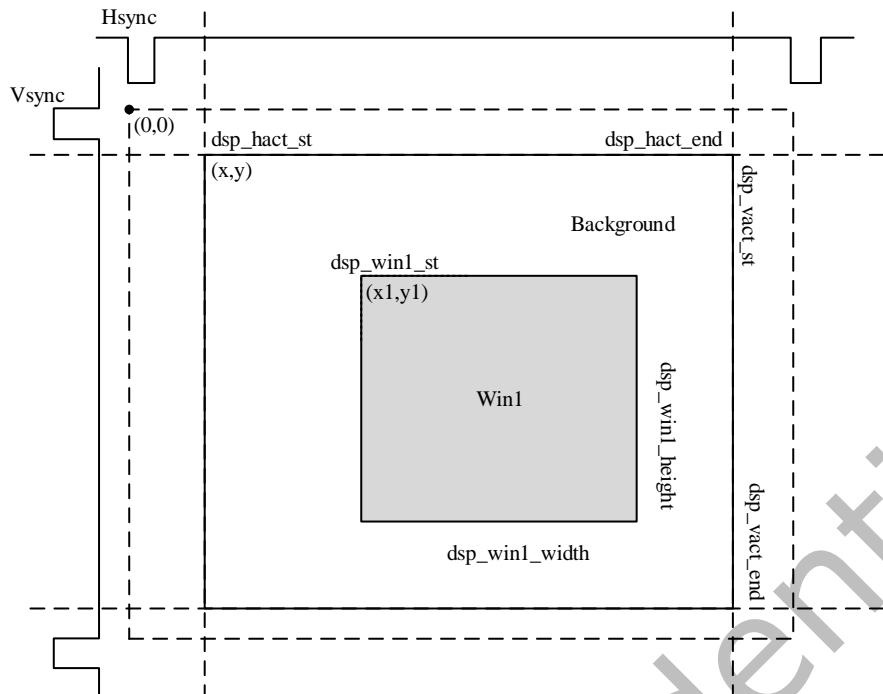


Fig. 5-6 VOP Overlay Display

2. Alpha Blending

Two blending modes are supported. One is per-pixel (ARGB) mode, the other is user-specified mode. In ARGB mode, the alpha value is in the ARGB data . In user-specified mode, the alpha value comes from the register (WINX_ALPHA_CTRL[11: 4]).

Pre-multiplied alpha are supported for per-pixel alpha in Win 1, for pre-multiplied alpha, the SRC data has already been multiplied with alpha value.

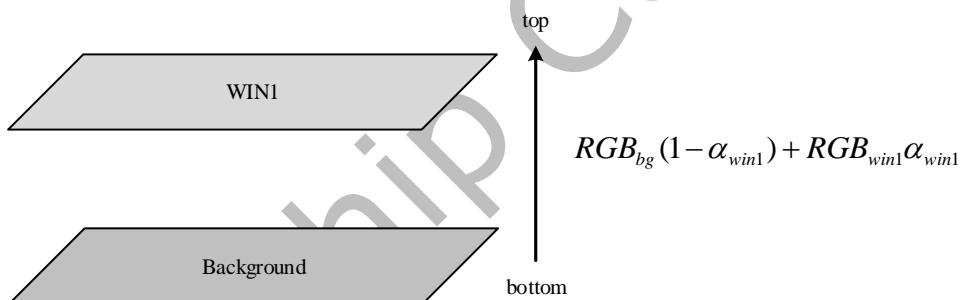


Fig. 5-7 VOP Alpha blending

5.3.5 BCSH

The BCSH block is used for brightness, contrast, saturation and hue adjustment to YCbCr format image.

The following diagram shows the BCSH adjustment processing. All the factors should be set in the VOP registers.

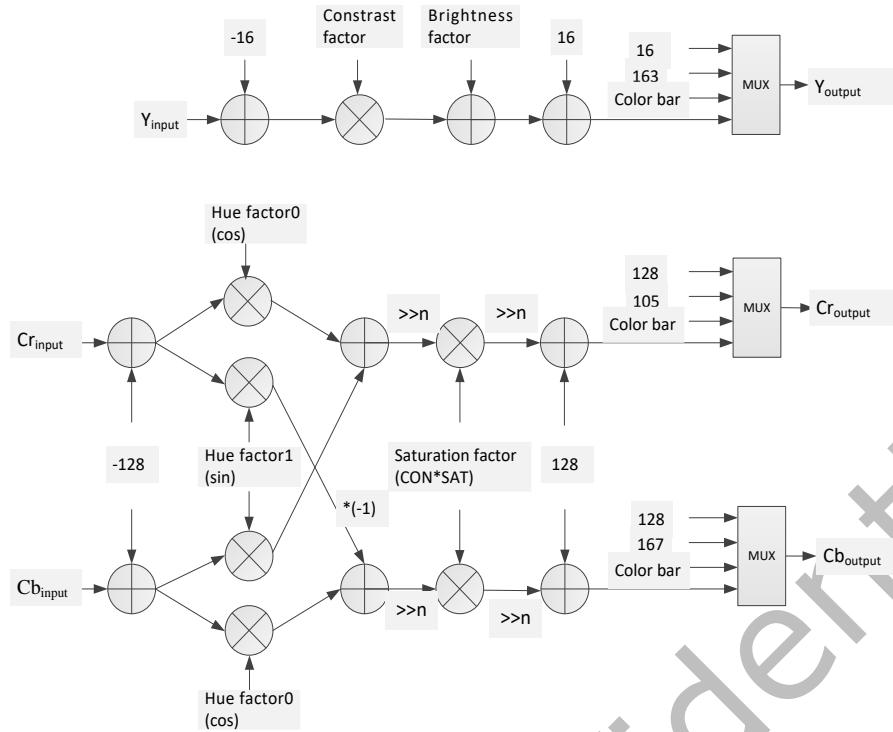


Fig. 5-8 VOP BCSH Diagram

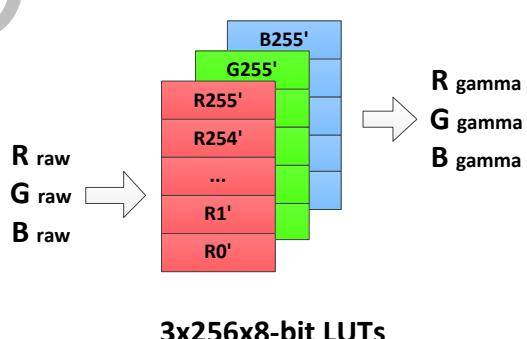
5.3.6 Gamma Correction

Gamma correction is necessary because most monitors don't have a linear relationship between the voltage and the brightness, which results in your scene looking like it has too much contrast and the light falling off from the source outward, happens too quickly. The result can also be problematic if you are going into a composition program.

You can correct this by "Gamma Correction", which allows you to display the images and textures on your computer in an accurate manner.

Your screen is not linear, in that it displays the brightness unevenly. As a result, the image looks to be more high contrast than it should, you end up adding more lights or turning up the intensity, or you don't use the lighting in a realistic way that matches well with live action scenes. It also creates problems for you if you use compositing software.

It consumes 256x8bit LUT for each channel. You can write gamma correction LUT through register "DSP_LUT_ADDR" one by one after set `dsp_lut_en = 0`.



3x256x8-bit LUTs

Fig. 5-9 VOP Gamma LUTs

5.3.7 Replication and dither down

1 Replication (dither up)

If the interface data bus is wider than the pixel format size, by programming the pixel components replication active/inactive, the MSB is replicated to the LSB of the interface data bus or the LSB is filled with 0s.

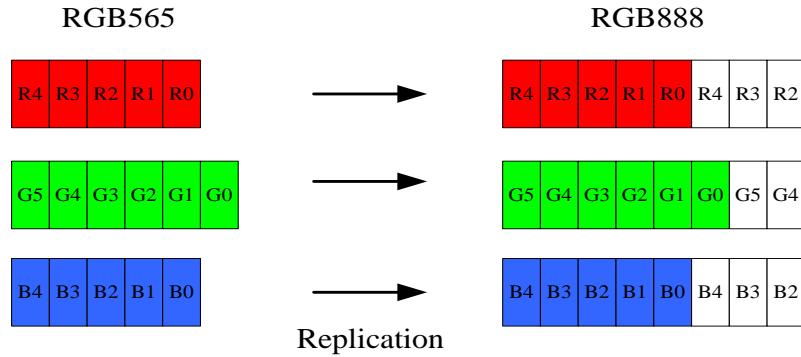


Fig. 5-10 VOP Replication

Dithering is used to improve the quality of display the pixel data in a lower color depth on the LCD panel. The Dithering algorithm is based on the (x,y) pixel position, the value of removed bits and frame counter.

2 Dither down

Here we support three kinds of dithering arithmetic. RGB888 to RGB666 has two ways, one is allegro, the other is FRC. RGB888 to RGB565 has only one arithmetic, which is allegro. When dithering is not enabled, the MSBs of the pixel color components are output on the interface data bus if the interface data bus is smaller than the pixel format size.

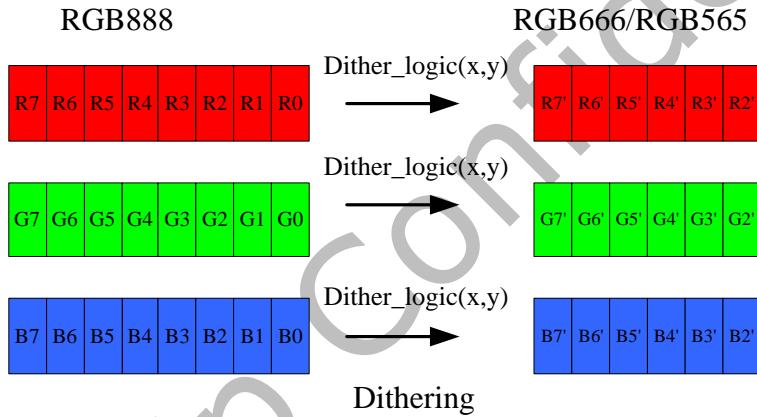


Fig. 5-11 VOP dithering

The following figure is the recommended pattern picture in vop, you can config different value of register 0x170~0x184, to change the pattern picture.(default value is recommended values below)

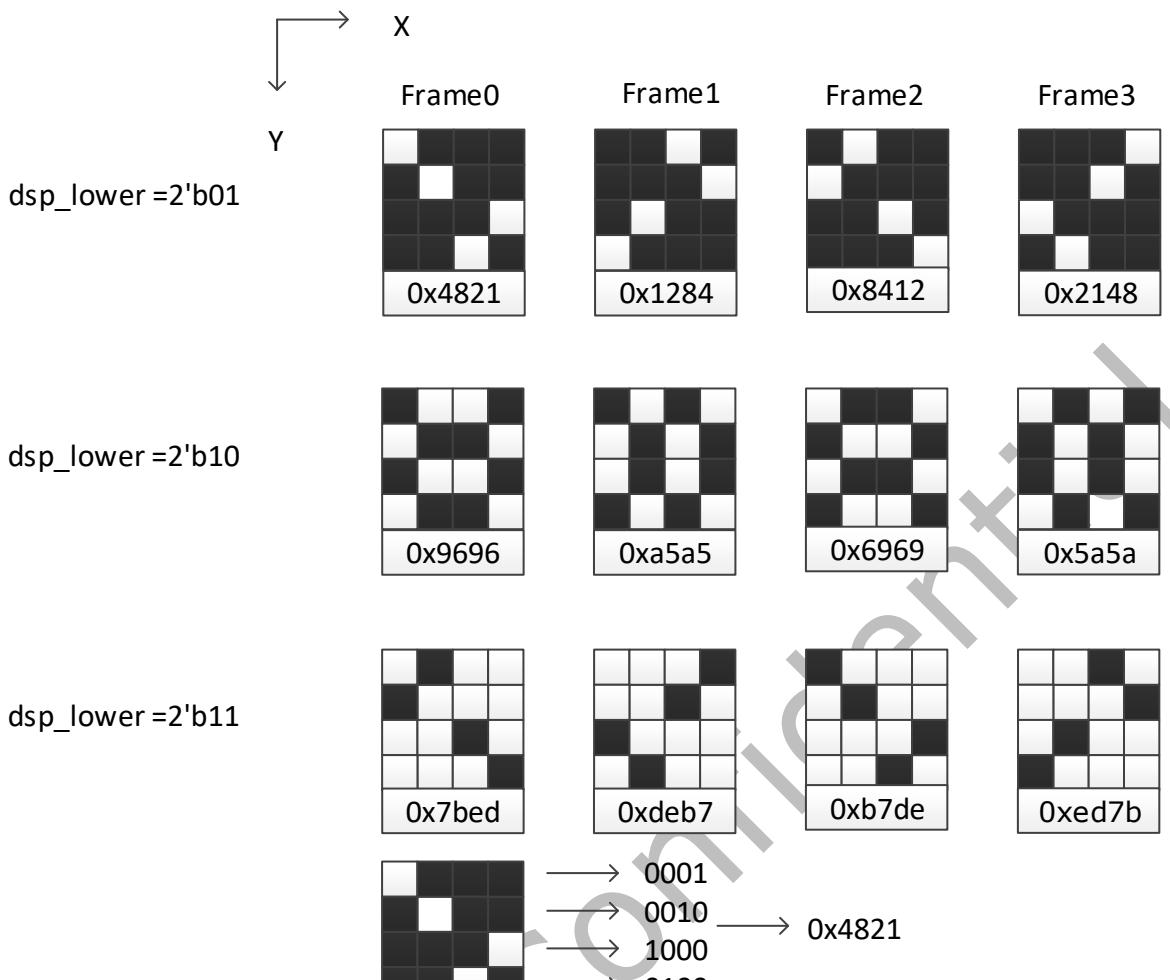


Fig. 5-12 FRC pattern diagram

Recommended pattern:

0x170 : 0x12844821
 0x174 : 0x21488412
 0x178 : 0xa55a9696
 0x17f : 0x5aa56969
 0x180 : 0xdeb77bed
 0x184 : 0xed7bb7de

5.4 Register Description

5.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

5.4.2 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|-------------------|--------|------|-------------|---------------------------|
| VOPS REG CFG DONE | 0x0000 | W | 0x00000000 | Register config done flag |
| VOPS VERSION | 0x0004 | W | 0x00000000 | Version for vop |
| VOPS DSP BG | 0x0008 | W | 0x00000000 | Display control register |
| VOPS MCU | 0x000c | W | 0x00000000 | MCU control register |
| VOPS SYS CTRL0 | 0x0010 | W | 0x00000000 | System control register |
| VOPS SYS CTRL1 | 0x0014 | W | 0x00000000 | System control register |
| VOPS SYS CTRL2 | 0x0018 | W | 0x00006000 | System control register |

| Name | Offset | Size | Reset Value | Description |
|--------------------------|--------|------|-------------|---|
| VOPS_DSP_CTRL0 | 0x0020 | W | 0x00000101 | Display register |
| VOPS_DSP_CTRL1 | 0x0024 | W | 0x00000000 | Display register |
| VOPS_DSP_CTRL2 | 0x0028 | W | 0x00004000 | Display register |
| VOPS_VOP_STATUS | 0x002c | W | 0x00000000 | Some vop module status |
| VOPS_LINE_FLAG | 0x0030 | W | 0x00000000 | Line flag config register |
| VOPS_INTR_EN | 0x0034 | W | 0x00000000 | Interrupt enable register |
| VOPS_INTR_CLEAR | 0x0038 | W | 0x00000000 | Interrupt clear register |
| VOPS_INTR_STATUS | 0x003c | W | 0x00000000 | Interrupt raw status and interrupt status |
| VOPS_WIN1_CTRL0 | 0x0090 | W | 0x00000000 | Win1 ctrl register |
| VOPS_WIN1_CTRL1 | 0x0094 | W | 0x00000500 | Win1 ctrl register |
| VOPS_WIN1_VIR | 0x0098 | W | 0x00000000 | Win1 virtual stride |
| VOPS_WIN1_MST | 0x00a0 | W | 0x00000000 | Win1 memory start address |
| VOPS_WIN1_DSP_INFO | 0x00a4 | W | 0x00ef013f | Win1 display width/height on panel |
| VOPS_WIN1_DSP_ST | 0x00a8 | W | 0x0000000a | Win1 display start point on panel |
| VOPS_WIN1_COLOR_KEY | 0x00ac | W | 0x00000000 | Win1 color key register |
| VOPS_WIN1_ALPHA_CTRL | 0x00bc | W | 0x00000000 | Blending control register |
| VOPS_DSP_HTOTAL_HS_END | 0x0100 | W | 0x014a000a | Panel scanning horizontal width and hsync pulse end point |
| VOPS_DSP_HACT_ST_EN_D | 0x0104 | W | 0x000a0000 | Panel active horizontal scanning start point and end point |
| VOPS_DSP_VTOTAL_VS-END | 0x0108 | W | 0x00fa000a | Panel scanning vertical height and vsync pulse end point |
| VOPS_DSP_VACT_ST_EN_D | 0x010c | W | 0x000a00fa | Panel active vertical scanning start point and end point |
| VOPS_DSP_VS_ST_END_F1 | 0x0110 | W | 0x00000000 | Vertical scanning start point and vsync pulse end point of even file |
| VOPS_DSP_VACT_ST_EN_D_F1 | 0x0114 | W | 0x00000000 | Vertical scanning active start point and end point of even filed in interlace |
| VOPS_BCSH_CTRL | 0x0160 | W | 0x00000000 | Brightness/Contrast enhancement/Saturation/Hue contrl |
| VOPS_BCSH_COL_BAR | 0x0164 | W | 0x00000000 | Video mode equals 2, then output color bar yuv 24bits value |
| VOPS_BCSH_BCS | 0x0168 | W | 0x00000000 | Brightness/Contrast enhancement/Saturation |
| VOPS_BCSH_H | 0x016c | W | 0x00000000 | Hue |
| VOPS_FRC_LOWER01_0 | 0x0170 | W | 0x12844821 | Frc algorithm configuration register |
| VOPS_FRC_LOWER01_1 | 0x0174 | W | 0x21488412 | Frc algorithm configuration register |

| Name | Offset | Size | Reset Value | Description |
|--------------------------|--------|------|-------------|---|
| VOPS_FRC_LOWER10_0 | 0x0178 | W | 0xa55a9696 | Frc algorithm configuration register |
| VOPS_FRC_LOWER10_1 | 0x017c | W | 0x5aa56969 | Frc algorithm configuration register |
| VOPS_FRC_LOWER11_0 | 0x0180 | W | 0xdeb77bed | Frc algorithm configuration register |
| VOPS_FRC_LOWER11_1 | 0x0184 | W | 0xed7bb7de | Frc algorithm configuration register |
| VOPS_MCU_RW_BYPASS_PORT | 0x018c | W | 0x00000000 | MCU panel write data |
| VOPS_DBG_REG_SCAN_LINE | 0x0190 | W | 0x00000000 | Current line number of dsp timing |
| VOPS_BLANKING_VALUE | 0x0194 | W | 0x00000000 | The value of vsync blanking |
| VOPS_FLAG_REG_FRM_VALID | 0x0198 | W | 0x00000000 | Flag reg value after frame valid |
| VOPS_FLAG_REG | 0x019c | W | 0x00000000 | Flag reg value before frame valid |
| VOPS_GAMMA_LUT_ADDR | 0x0a00 | W | 0x00000000 | note: SIZE: 24X256 used for panel GAMMA adjustment, base address: 0x0a00 -- 0xdff |
| VOPS_MMU_DTE_ADDR | 0x0f00 | W | 0x00000000 | MMU current page Table address |
| VOPS_MMU_STATUS | 0x0f04 | W | 0x00000000 | MMU status register |
| VOPS_MMU_COMMAND | 0x0f08 | W | 0x00000000 | MMU command register |
| VOPS_MMU_PAGE_FAULT_ADDR | 0x0f0c | W | 0x00000000 | MMU logical address of last page fault |
| VOPS_MMU_ZAP_ONE_LINE | 0x0f10 | W | 0x00000000 | MMU Zap cache line register |
| VOPS_MMU_INT_RAWSTATUS | 0x0f14 | W | 0x00000000 | MMU raw interrupt status register |
| VOPS_MMU_INT_CLEAR | 0x0f18 | W | 0x00000000 | MMU raw interrupt status register |
| VOPS_MMU_INT_MASK | 0x0f1c | W | 0x00000000 | MMU raw interrupt status register |
| VOPS_MMU_INT_STATUS | 0x0f20 | W | 0x00000000 | MMU raw interrupt status register |
| VOPS_MMU_AUTO_GATING | 0x0f24 | W | 0x00000000 | mmu auto gating |
| VOPS_MMU_CFG_DONE | 0x0f28 | W | 0x00000000 | mmu config done reg |

Notes: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

5.4.3 Detail Register Description

VOPS_REG_CFG_DONE

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | write_mask When every bit HIGH, enable the writing corresponding bit When every bit LOW, don't care the writing corresponding bit |
| 15:6 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 5 | RW | 0x0 | <p>reg_load_sys_en In the first setting of the register, the new value was saved into the mirror register.</p> <p>When all the system register config finish(all reg except win1), writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame</p> |
| 4 | RW | 0x0 | <p>reg_load_iep_en In the first setting of the register, the new value was saved into the mirror register.</p> <p>When all the iep register config finish(only 2 signals direct_path_en,direct_path_layer_sel), writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame</p> |
| 3 | RW | 0x0 | reserved |
| 2 | RW | 0x0 | <p>reg_load_win1_en In the first setting of the register, the new value was saved into the mirror register.</p> <p>When all the win1 register config finish, writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame</p> |
| 1 | RW | 0x0 | reserved |
| 0 | WO | 0x0 | <p>reg_load_global_en In the first setting of the register, the new value was saved into the mirror register.</p> <p>When all the register configured finish, writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame</p> |

VOPS VERSION

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | major used for IP structure |
| 23:16 | RO | 0x00 | minor big feature change under same structure |
| 15:0 | RO | 0x0000 | build rtl current svn number |

VOPS DSP BG

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:24 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 23:16 | RW | 0x00 | dsp_bg_red Background Red color |
| 15:8 | RW | 0x00 | dsp_bg_green Background Green color |
| 7:0 | RW | 0x00 | dsp_bg_blue Background Blue color |

VOPS MCU

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | mcu_type MCU LCD output SELECT |
| 30 | RW | 0x0 | mcu_bypass MCU LCD BYPASS MODE Select |
| 29 | RW | 0x0 | mcu_rs MCU LCD RS Select 1'b0: Command 1'b1: Data |
| 28 | RW | 0x0 | mcu_frame_st Write"1": MCU HOLD Mode Frame Start Read: MCU/LCDC standby HOLD status |
| 27 | RW | 0x0 | mcu_hold_mode MCU HOLD Mode Select |
| 26 | RW | 0x0 | mcu_clk_sel 1'b0: MCU BYPASS sync with HCLK 1'b1: MCU BYPASS sync with DCLK |
| 25:20 | RW | 0x00 | mcu_rw_pend MCU_RW signal end point (0-63) |
| 19:16 | RW | 0x0 | mcu_rw_pst MCU_RW signal start point (0-15) |
| 15:10 | RW | 0x00 | mcu_cs_pend MCU_CS signal end point (0-63) |
| 9:6 | RW | 0x0 | mcu_cs_pst MCU_CS signal start point (0-15) |
| 5:0 | RW | 0x00 | mcu_pix_total MCU LCD Interface writing period (1-63) |

VOPS SYS CTRL0

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:0 | RW | 0x00000000 | reserved Reserved |

VOPS SYS CTRL1

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | sw_axi_max_outstand_num axi max outstanding number |
| 15:13 | RO | 0x0 | reserved |
| 12 | RW | 0x0 | sw_axi_max_outstand_en 1'b0: Disable 1'b1: Enable |
| 11:8 | RW | 0x0 | sw_noc_hurry_threshold noc hurry threshold |
| 7 | RO | 0x0 | reserved |
| 6:5 | RW | 0x0 | sw_noc_hurry_value noc hurry value |
| 4 | RW | 0x0 | sw_noc_hurry_en 1'b0: Disable 1'b1: Enable |
| 3 | RO | 0x0 | reserved |
| 2:1 | RW | 0x0 | sw_noc_qos_value noc qos value |
| 0 | RW | 0x0 | sw_noc_qos_en 1'b0: Disable 1'b1: Enable |

VOPS SYS CTRL2

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15 | RW | 0x0 | dcf_idle_en 1'b0: Dcf idle disable 1'b1: Dcf idle enable |
| 14 | RW | 0x1 | fs_addr_mask_en 1'b0: Disable 1'b1: Enable |
| 13 | RW | 0x1 | imd_global_regdone_en 1'b0: DISABLE 1'b1: ENALBE |
| 12 | RW | 0x0 | imd_dsp_timing_imd 1'b0: Timing reg valid immediatly 1'b1: Timing reg valid after frame start |
| 11 | RW | 0x0 | imd_edpi_frm_st wms_fs |
| 10 | RW | 0x0 | imd_edpi_ctrl_mode 1'b0: Wms is enable by te and wms_fs 1'b1: Wms is enable by te only |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 9 | RW | 0x0 | imd_edpi_te_en 1'b0: Disable TE 1'b1: Enable TE |
| 8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | sw_io_pad_clk_sel 1'b0: Normal dclk out 1'b1: Gating dclk out |
| 6 | RW | 0x0 | imd_dsp_data_out_mode 1'b0: Normal output mode 1'b1: Output 24'b0 |
| 5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | imd_yuv_clip 1'b0: Disable, YCbCr no clip 1'b1: Enable, YCbCr clip before YCbCr2RGB *Y clip: 16~235, CBCR clip: 16~239 |
| 3 | RW | 0x0 | imd_dsp_out_zero 1'b0: Normal output 1'b1: Output '0' means:hsync,vsync,den =000 |
| 2 | RW | 0x0 | imd_vop_dma_stop 1'b0: Disable 1'b1: Enable *If DMA is working, the stop mode would not be active until current bus transfer is finished |
| 1 | RW | 0x0 | imd_vop_standby_en Writing "1" to turn VOP into standby mode, All the layer would disable and the data transfer from frame buffer memory would stop at the end of current frame. The output would be blank. When writing "0" to this bit, standby mode would disable and the VOP go back to work immediately. 1'b0: Disable 1'b1: Enable *Black display is recommended before setting standby mode enable |
| 0 | RW | 0x0 | imd_auto_gating_en 1'b0: Disable 1'b1: Enable |

VOPS DSP CTRL0

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|-------------|
| 31:29 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 28 | RW | 0x0 | mipi_den_pol 1'b0: Positive 1'b1: Negative when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 27 | RW | 0x0 | mipi_vsync_pol 1'b0: Negative 1'b1: Positive when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 26 | RW | 0x0 | mipi_hsync_pol 1'b0: Negative 1'b1: Positive when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 25 | RW | 0x0 | mipi_dclk_pol 1'b0: Mipi dclk inv disable 1'b1: Mipi dclk inv enable |
| 24 | RW | 0x0 | mipi_dclk_en 1'b0: Mipi dclk disable 1'b1: Mipi dclk enable |
| 23:15 | RO | 0x0 | reserved |
| 14 | RW | 0x0 | sw_hdmi_clk_i_sel 1'b0: Select dclk to hdmi io 1'b1: Select dclk_core to hdmi io |
| 13 | RW | 0x0 | sw_core_clk_sel 1'b0: Select dclk sclk 1'b1: Select dclk_div sclk_div |
| 12 | RW | 0x0 | hdmi_den_pol 1'b0: Positive 1'b1: Negative when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 11 | RW | 0x0 | hdmi_vsync_pol 1'b0: Negative 1'b1: Positive when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 10 | RW | 0x0 | hdmi_hsync_pol 1'b0: Negative 1'b1: Positive when this channel dclk_en = 0, this signal used for contrl corresponding bit |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9 | RW | 0x0 | hdmi_dclk_pol 1'b0: Hdmi dclk inv disable 1'b1: Hdmi dclk inv enable |
| 8 | RW | 0x1 | hdmi_dclk_en 1'b0: Hdmi dclk disable 1'b1: Hdmi dclk enable |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | rgb_den_pol 1'b0: Positive 1'b1: Negative when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 3 | RW | 0x0 | rgb_vsync_pol 1'b0: Negative 1'b1: Positive when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 2 | RW | 0x0 | rgb_hsync_pol 1'b0: Negative 1'b1: Positive when this channel dclk_en = 0, this signal used for contrl corresponding bit |
| 1 | RW | 0x0 | rgb_dclk_pol 1'b0: Rgb dclk inv disable 1'b1: Rgb dclk inv enable |
| 0 | RW | 0x1 | rgb_dclk_en 1'b0: Rgb dclk disable 1'b1: Rgb dclk enable |

VOPS DSP CTRL1

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:0 | RW | 0x00000000 | reserved Reserved |

VOPS DSP CTRL2

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:20 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 19:16 | RW | 0x0 | <p>dsp_out_mode</p> <p>4'b0000: Parallel 24-bit RGB888 output {R[7:0],G[7:0],B[7:0]}</p> <p>4'b0001: Parallel 18-bit RGB666 output {6'b0,R[5:0],G[5:0],B[5:0]}</p> <p>4'b0010: Parallel 16-bit RGB565 output {8'b0,R[4:0],G[5:0],B[4:0]}</p> <p>Others: Reserved</p> |
| 15 | RW | 0x0 | <p>dsp_black_en</p> <p>When this bit enable, the pixel data output is all black (0x000000)</p> |
| 14 | RW | 0x1 | <p>dsp_blank_en</p> <p>When this bit enable, the hsync/vsync/den output is blank. means:hsync,vsync,den =110</p> |
| 13 | RO | 0x0 | reserved |
| 12 | RW | 0x0 | <p>dsp_rg_swap</p> <p>1'b0: RGB</p> <p>1'b1: GRB</p> |
| 11 | RW | 0x0 | <p>dsp_rb_swap</p> <p>1'b0: RGB</p> <p>1'b1: BGR</p> |
| 10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | <p>dsp_bg_swap</p> <p>1'b0: RGB</p> <p>1'b1: RBG</p> |
| 8 | RW | 0x0 | <p>dither_down</p> <p>1'b0: Disable</p> <p>1'b1: Enable</p> |
| 7 | RW | 0x0 | <p>dither_down_sel</p> <p>1'b0: Allegro</p> <p>1'b1: FRC</p> |
| 6 | RW | 0x0 | <p>dither_down_mode</p> <p>1'b0: RGB888 to RGB565</p> <p>1'b1: RGB888 to RGB666</p> |
| 5 | RW | 0x0 | <p>dsp_lut_en</p> <p>1'b0: Disable</p> <p>1'b1: Enable</p> <p>*This bit should be "0" when CPU updates the LUT, and should be "1" when Display LUT mode enable</p> |
| 4 | RW | 0x0 | <p>sw_overlay_mode</p> <p>1'b0: Overlay in rgb domain</p> <p>1'b1: Overlay in yuv domain</p> |
| 3 | RW | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 2 | RW | 0x0 | dither_up 1'b0: Disable 1'b1: Enable |
| 1 | RW | 0x0 | reserved |
| 0 | RO | 0x0 | reserved |

VOPS VOP STATUS

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | reserved |
| 4 | RO | 0x0 | dma_stop_valid vop dma stop status |
| 3 | RO | 0x0 | reserved |
| 2 | RO | 0x0 | int_raw_dma_finish dma finish raw signal 1'b0: Not finish 1'b1: Finish |
| 1 | RO | 0x0 | idle_mmu_ff1 mmu idle status 1'b0: Busy 1'b1: Idle |
| 0 | RO | 0x0 | dsp_blinking_en_async_aff2 1'b0: Blanking disable 1'b1: Blanking enable |

VOPS LINE FLAG

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x000 | dsp_line_flag1_num The display line number when the flag interrupt 1 occur, the range is (0~ DSP_VTOTAL-1) |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x000 | dsp_line_flag0_num The display line number when the flag interrupt occur, the range is (0~ DSP_VTOTAL-1) |

VOPS INTR EN

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | write_mask When every bit HIGH, enable the writing corresponding bit When every bit LOW, don't care the writing corresponding bit |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 15 | RW | 0x0 | mmu_irq_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 14:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | dma_frm_fsh_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 8 | RW | 0x0 | dsp_hold_valid_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 7 | RW | 0x0 | win1_empty_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 6 | RW | reserved | reserved |
| 5 | RW | 0x0 | bus_error_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 4 | RW | 0x0 | line_flag1_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 3 | RW | 0x0 | line_flag0_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 2 | RW | 0x0 | addr_same_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 1 | RW | 0x0 | fs1_intr_en 1'b0: DISABLE 1'b1: ENABLE |
| 0 | RW | 0x0 | fs0_intr_en 1'b0: DISABLE 1'b1: ENABLE |

VOPS INTR CLEAR

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15 | RW | 0x0 | mmu_irq_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 14:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | dma_frm_fsh_intr_clr 1'b0: DISABLE 1'b1: ENABLE |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 8 | W1 C | 0x0 | dsp_hold_valid_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 7 | W1 C | 0x0 | win1_empty_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 6 | W1 C | 0x0 | reserved |
| 5 | W1 C | 0x0 | bus_error_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 4 | W1 C | 0x0 | line_flag1_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 3 | W1 C | 0x0 | line_flag0_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 2 | W1 C | 0x0 | addr_same_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 1 | W1 C | 0x0 | fs1_intr_clr 1'b0: DISABLE 1'b1: ENABLE |
| 0 | W1 C | 0x0 | fs0_intr_clr 1'b0: DISABLE 1'b1: ENABLE |

VOPS INTR STATUS

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | mmu_intr_raw_status mmu interrupt raw status |
| 30:26 | RO | 0x0 | reserved |
| 25 | RO | 0x0 | dma_frm_fsh_intr_raw_sts dma finish interrupt raw status |
| 24 | RO | 0x0 | dsp_hold_valid_intr_raw_sts display hold interrupt raw status |
| 23 | RO | 0x0 | win1_empty_intr_raw_sts win1 empty interrupt raw status |
| 22 | RO | 0x0 | reserved |
| 21 | RO | 0x0 | bus_error_intr_raw_sts bus error interrupt raw status |
| 20 | RO | 0x0 | line_flag1_intr_raw_sts line flag1 interrupt raw status |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 19 | RO | 0x0 | line_flag0_intr_raw_sts line flag0 interrupt raw status |
| 18 | RW | 0x0 | addr_same_intr_raw_sts same address interrupt raw status |
| 17 | RO | 0x0 | fs1_intr_raw_sts new frame start interrupt raw status |
| 16 | RO | 0x0 | fs0_intr_raw_sts frame start interrupt raw status |
| 15 | RO | 0x0 | mmu_intr_status mmu interrupt status |
| 14:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | dma_frm_fsh_intr_sts dma finish interrupt status |
| 8 | RO | 0x0 | dsp_hold_valid_intr_sts display hold interrupt status |
| 7 | RO | 0x0 | win1_empty_intr_sts win1 empty interrupt status |
| 6 | RO | 0x0 | reserved |
| 5 | RO | 0x0 | bus_error_intr_sts bus error interrupt status |
| 4 | RO | 0x0 | line_flag1_intr_sts line flag1 interrupt status |
| 3 | RO | 0x0 | line_flag0_intr_sts line flag0 interrupt status |
| 2 | RW | 0x0 | addr_same_intr_sts same address interrupt status |
| 1 | RW | 0x0 | fs1_intr_sts new frame start interrupt status |
| 0 | RO | 0x0 | fs0_intr_sts frame start interrupt status |

VOPS WIN1 CTRL0

Address: Operational Base + offset (0x0090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | win1_alpha_swap 1'b0: ARGB 1'b1: RGBA |
| 12 | RW | 0x0 | win1_rb_swap 1'b0: RGB 1'b1: BGR |
| 11:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | win1_no_outstanding 1'b0: Enable 1'b1: Disable |
| 8 | RW | 0x0 | win1_interlace_read 1'b0: Disable 1'b1: Enable |
| 7 | RO | 0x0 | reserved |
| 6:4 | RW | 0x0 | win1_data_fmt 3'b000: ARGB888 3'b001: RGB888 3'b010: RGB565 others: Reserved |
| 3:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | win1_en 1'b0: Win1 layer disable 1'b1: Win1 layer enable |

VOPS WIN1 CTRL1

Address: Operational Base + offset (0x0094)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x5 | sw_win1_rid win1 axi read id |
| 7:4 | RW | 0x0 | win1_axi_gather_num win1 axi gather_num gather number |
| 3:2 | RW | 0x0 | win1_dma_burst_length 2'b00: Burst16 (burst 15 in rgb888 pack mode) 2'b01: Burst8 (burst 12 in rgb888 pack mode) 2'b10: Burst4 (burst 6 in rgb888 pack mode) |
| 1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | win1_axi_gather_en 1'b0: Disable 1'b1: Enable |

VOPS WIN1 VIR

Address: Operational Base + offset (0x0098)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | win1_vir_stride Number of words of Win1 Virtual width ARGB888: Win1_vir_width RGB888: (Win1_vir_width*3/4) + (win1_vir_width%3) RGB565: Ceil(win1_vir_width/2) |

VOPS WIN1 MST

Address: Operational Base + offset (0x00a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:0 | RW | 0x00000000 | win1_mst |

VOPS WIN1 DSP INFO

Address: Operational Base + offset (0x00a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x0 | reserved |
| 26:16 | RW | 0x0ef | dsp_win1_height win1_dsp_height = (win1 dsp vertical size -1) |
| 15:11 | RO | 0x0 | reserved |
| 10:0 | RW | 0x13f | dsp_win1_width win1_dsp_width = (win1 dsp horizontal size -1) |

VOPS WIN1 DSP ST

Address: Operational Base + offset (0x00a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x000 | dsp_win1_yst win1 y-axis start point |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x00a | dsp_win1_xst win1 x-axis start point |

VOPS WIN1 COLOR KEY

Address: Operational Base + offset (0x00ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | win1_key_en 1'b0: Disable 1'b1: Enable |
| 23:0 | RW | 0x000000 | win1_key_color win1 color key |

VOPS WIN1 ALPHA CTRL

Address: Operational Base + offset (0x00bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | reserved |
| 11:4 | RW | 0x00 | win1_alpha_value win1 global alpha value |
| 3 | RW | 0x0 | win1_alpha_sat_mode 1'b0: Alpha value no change 1'b1: Alpha = alpha + alpha[7] |
| 2 | RW | 0x0 | win1_alpha_pre_mul 1'b0: Non-premultiplied alpha 1'b1: Premultiplied alpha |
| 1 | RW | 0x0 | win1_alpha_mode 1'b0: User-defined alpha 1'b1: Per-pixel alpha |
| 0 | RW | 0x0 | win1_alpha_en 1'b0: Disable 1'b1: Enable |

VOPS DSP HTOTAL HS END

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x14a | dsp_htotal dsp_htotal |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x00a | dsp_hs_end dsp_hs_end |

VOPS DSP HACT ST END

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------------|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x00a | dsp_hact_st dsp_hact_st |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x000 | dsp_hact_end dsp_hact_end |

VOPS DSP VTOTAL VS END

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x0fa | dsp_vtotal dsp_vtotal |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x00a | dsp_vs_end dsp_vs_end |

VOPS DSP VACT ST END

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------------|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x00a | dsp_vact_st dsp_vact_st |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x0fa | dsp_vact_end dsp_vact_end |

VOPS DSP VS ST END F1

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x000 | dsp_vs_st_f1 Panel display scanning vertical vsync start point of 2nd field (interlace display mode) |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x000 | dsp_vs_end_f1 Panel display scanning vertical vsync end point of 2nd field(interlace display mode) |

VOPS DSP VACT ST END F1

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:16 | RW | 0x000 | dsp_vact_st_f1 Panel display scanning vertical active start point of 2nd field (interlace display mode) |
| 15:12 | RO | 0x0 | reserved |
| 11:0 | RW | 0x000 | dsp_vact_end_f1 Panel display scanning vertical active end point of 2nd field (interlace display mode) |

VOPS BCSH CTRL

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | sw_bcsh_r2y_en 1'b0: Bypass 1'b1: Enable |
| 6 | RW | 0x0 | sw_bcsh_y2r_en 1'b0: Bypass 1'b1: Enable |
| 5:4 | RW | 0x0 | sw_bcsh_y2r_csc_mode Color space conversion: 2'b00/11: Mpeg 2'b01: Hd 2'b10: Jpeg |
| 3:2 | RW | 0x0 | video_mode 2'b00: Black 2'b01: Blue 2'b10: Color bar 2'b11: Normal video |
| 1 | RW | 0x0 | sw_bcsh_r2y_csc_mode Color space conversion: 1'b0: BT601 1'b1: BT709 |
| 0 | RW | 0x0 | bcsh_en 1'b0: Bcsh bypass 1'b1: Bcsh enable |

VOPS BCSH COL BAR

Address: Operational Base + offset (0x0164)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x00 | color_bar_v v component of color bar |
| 15:8 | RW | 0x00 | color_bar_u u component of color bar |
| 7:0 | RW | 0x00 | color_bar_y y component of color bar |

VOPS BCSH BCS

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29:20 | RW | 0x000 | sat_con Saturation*Contrast*256: 0,1.992*1.992 |
| 19:17 | RO | 0x0 | reserved |
| 16:8 | RW | 0x000 | contrast Contrast*256: 0,1.992 |
| 7 | RO | 0x0 | reserved |
| 6:0 | RW | 0x00 | brightness Brightness: -128,127 |

VOPS BCSH H

Address: Operational Base + offset (0x016c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:25 | RO | 0x0 | reserved |
| 24:16 | RW | 0x000 | cos_hue cos_hue |
| 15:9 | RO | 0x0 | reserved |
| 8:0 | RW | 0x000 | sin_hue sin_hue |

VOPS FRC LOWER01_0

Address: Operational Base + offset (0x0170)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------------|
| 31:16 | RW | 0x1284 | lower01_frm1 lower01_frm1 |
| 15:0 | RW | 0x4821 | lower01_frm0 lower01_frm0 |

VOPS FRC LOWER01_1

Address: Operational Base + offset (0x0174)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------------|
| 31:16 | RW | 0x2148 | lower01_frm3 lower01_frm3 |
| 15:0 | RW | 0x8412 | lower01_frm2 lower01_frm2 |

VOPS FRC LOWER10_0

Address: Operational Base + offset (0x0178)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------------|
| 31:16 | RW | 0xa55a | lower10_frm1 lower10_frm1 |
| 15:0 | RW | 0x9696 | lower10_frm0 lower10_frm0 |

VOPS FRC LOWER10_1

Address: Operational Base + offset (0x017c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|------------------------------|
| 31:16 | RW | 0x5aa5 | lower10_frm3 lower10_frm3 |
| 15:0 | RW | 0x6969 | lower10_frm2 lower10_frm2 |

VOPS FRC LOWER11_0

Address: Operational Base + offset (0x0180)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|------------------------------|
| 31:16 | RW | 0xdeb7 | lower11_frm1 lower11_frm1 |
| 15:0 | RW | 0x7bed | lower11_frm0 lower11_frm0 |

VOPS FRC LOWER11_1

Address: Operational Base + offset (0x0184)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|------------------------------|
| 31:16 | RW | 0xed7b | lower11_frm3 lower11_frm3 |
| 15:0 | RW | 0xb7de | lower11_frm2 lower11_frm2 |

VOPS MCU RW BYPASS PORT

Address: Operational Base + offset (0x018c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | mcu_write_data_bypass mcu write bypass data |

VOPS DBG REG SCAN LINE

Address: Operational Base + offset (0x0190)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---------------|
| 31:12 | RO | 0x0 | reserved |
| 11:0 | RO | 0x000 | scan_line_num |

VOPS BLANKING VALUE

Address: Operational Base + offset (0x0194)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | blanking_value_config_en 1'b0: Disable 1'b1: Enable |
| 23:0 | RW | 0x0000000 | sw_blank_value blank value |

VOPS FLAG REG FRM VALID

Address: Operational Base + offset (0x0198)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x000000000 | flag_reg_frm_valid valid by frame start |

VOPS FLAG REG

Address: Operational Base + offset (0x019c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:0 | WO | 0x000000000 | flag_reg |

VOPS GAMMA LUT ADDR

Address: Operational Base + offset (0x0a00)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:24 | RO | 0x0 | reserved |
| 23:0 | RW | 0x0000000 | gamma_lut_addr |

VOPS MMU DTE ADDR

Address: Operational Base + offset (0x0f00)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------|
| 31:0 | RW | 0x000000000 | MMU_DTE_ADDR MMU DTE address |

VOPS MMU STATUS

Address: Operational Base + offset (0x0f04)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | reserved |
| 9:5 | RW | 0x00 | PAGE_FAULT_BUS_ID Index of master responsible for last page fault |
| 4 | RW | 0x0 | PAGE_FAULT_IS_WRITE The direction of access for last page fault: 1'b0 = Read 1'b1 = Write |
| 3 | RW | 0x0 | MMU_IDLE The MMU is idle when accesses are being translated and there are no unfinished translated accesses |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 2 | RW | 0x0 | STAIL_ACTIVE MMU stall mode currently enabled. The mode is enabled by command |
| 1 | RW | 0x0 | PAGE_FAULT_ACTIVE MMU page fault mode currently enabled. The mode is enabled by command |
| 0 | RW | 0x0 | PAGING_ENABLED Paging is enabled |

VOPS_MMU_COMMAND

Address: Operational Base + offset (0x0f08)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x0 | MMU_CMD MMU_CMD. This can be: 3'b000: MMU_ENABLE_PAGING 3'b001: MMU_DISABLE_PAGING 3'b010: MMU_ENABLE_STALL 3'b011: MMU_DISABLE_STALL 3'b100: MMU_ZAP_CACHE 3'b101: MMU_PAGE_FAULT_DONE 3'b110: MMU_FORCE_RESET |

VOPS_MMU_PAGE_FAULT_ADDR

Address: Operational Base + offset (0x0f0c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | PAGE_FAULT_ADDR address of last page fault |

VOPS_MMU_ZAP_ONE_LINE

Address: Operational Base + offset (0x0f10)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | MMU_ZAP_ONE_LINE address to be invalidated from the page table cache |

VOPS_MMU_INT_RAWSTAT

Address: Operational Base + offset (0x0f14)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------------------|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR read bus error |
| 0 | RW | 0x0 | PAGE_FAULT page fault |

VOPS_MMU_INT_CLEAR

Address: Operational Base + offset (0x0f18)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|----------------------------------|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR read bus error |
| 0 | RW | 0x0 | PAGE_FAULT page fault |

VOPS_MMU_INT_MASK

Address: Operational Base + offset (0x0f1c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|----------------------------------|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR read bus error |
| 0 | RW | 0x0 | PAGE_FAULT page fault |

VOPS_MMU_INT_STATUS

Address: Operational Base + offset (0x0f20)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|----------------------------------|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | READ_BUS_ERROR read bus error |
| 0 | RW | 0x0 | PAGE_FAULT page fault |

VOPS_MMU_AUTO_GATING

Address: Operational Base + offset (0x0f24)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | sw_mmu_cfg_mode when it is 1'b0, the mmu reg config will valid after frame start |
| 0 | RW | 0x0 | mmu_auto_gating when it is 1'b1, the mmu will auto gating it self |

VOPS_MMU_CFG_DONE

Address: Operational Base + offset (0x0f28)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | reg_load_mmu_en In the first setting of the register, the new value was saved into the mirror register. When all the mmu register config finish, writing this register to enable the copyright of the mirror register to real register |

5.5 Timing Diagram

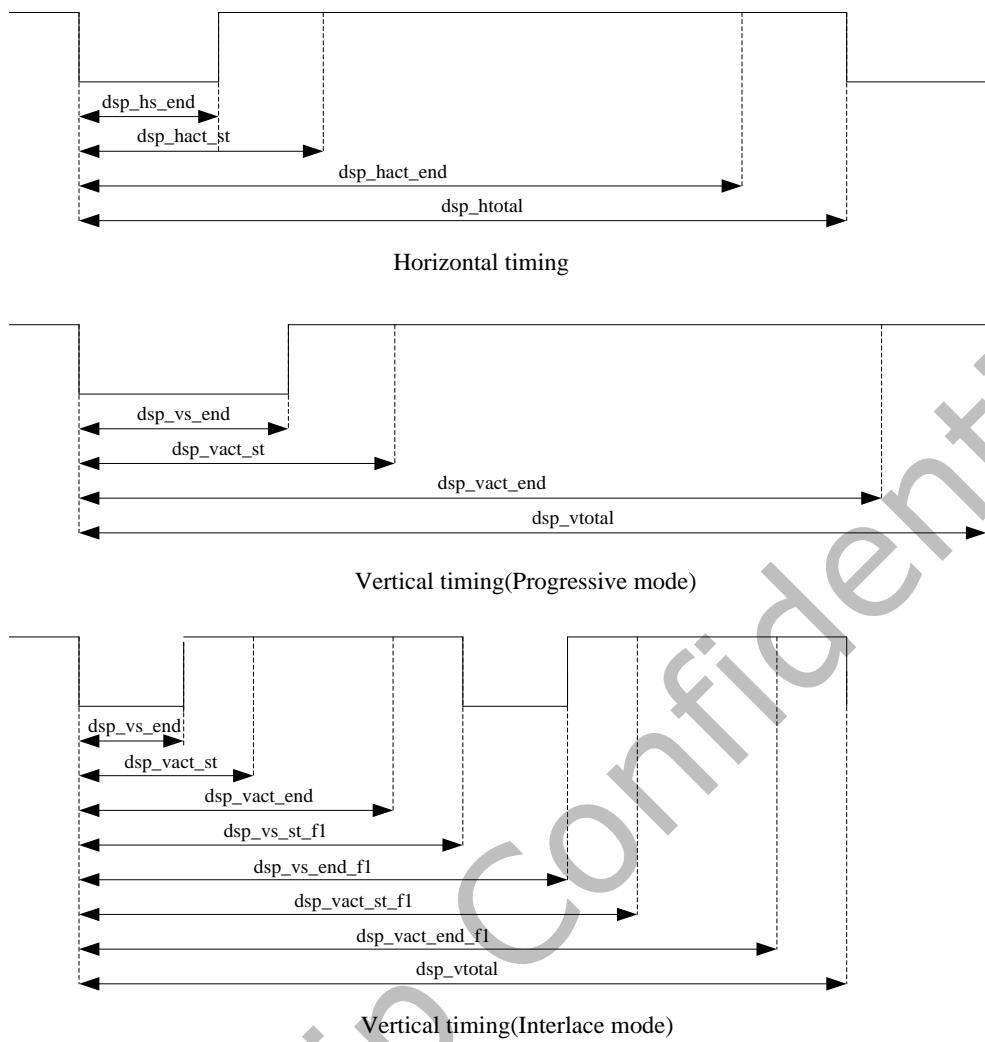


Fig. 5-13 VOP RGB interface timing setting

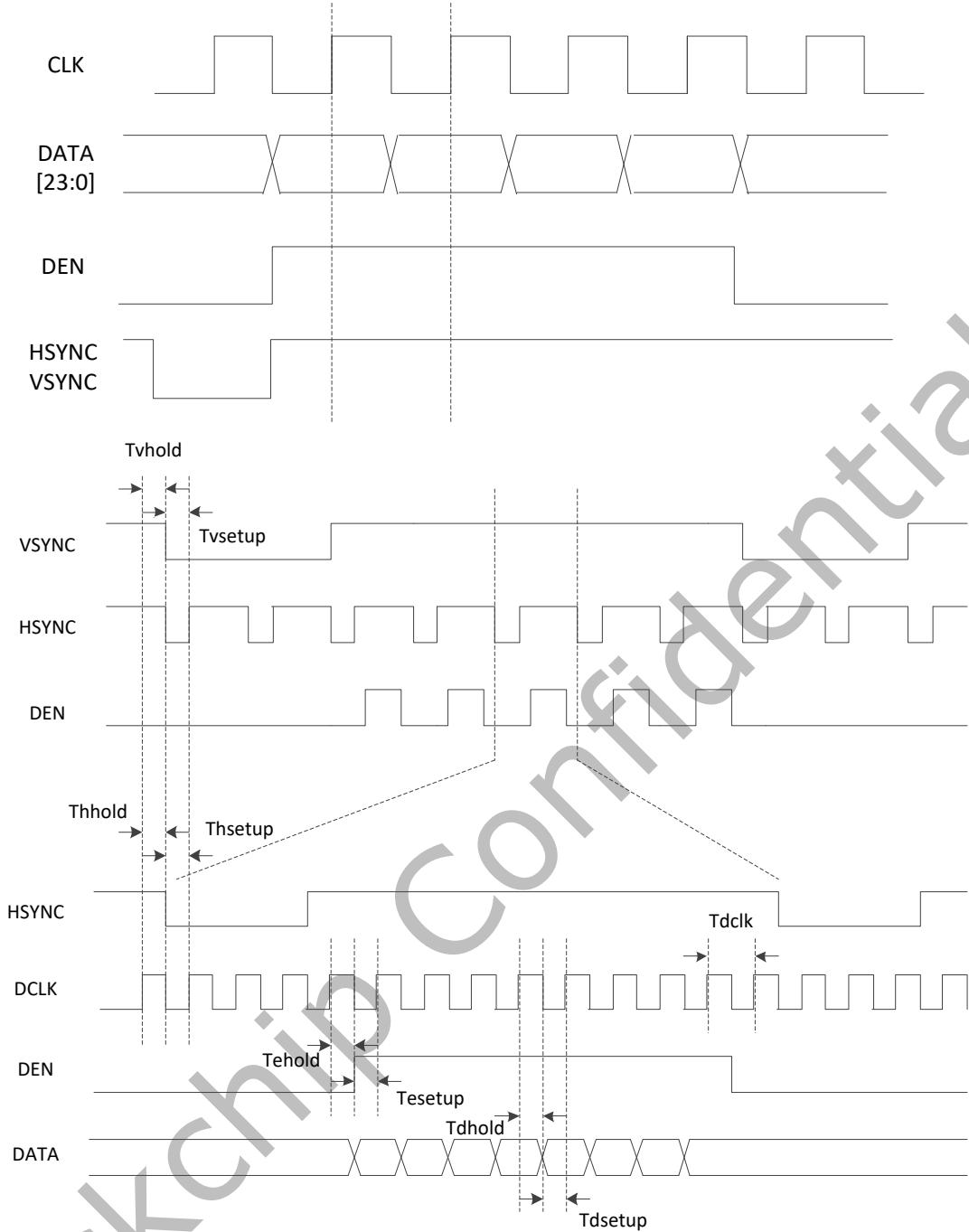


Fig. 5-14 VOP RGB Interface Timing(SDR)

5.6 Interface Description

5.6.1 VOP Outputs

VOP supports RGB, MIPI, output. the MIPI output interface is same as RGB interface. VOP is suitable for different display mode by different usage, which is shown as follows.

Table 5-2 VOP Control Pins Definition

| Display mode | RGB Parallel 18-bit | RGB Parallel 16-bit |
|--------------|---------------------|---------------------|
| DCLK | DCLK | DCLK |
| VSYNC | VSYNC | VSYNC |
| HSYNC | HSYNC | HSYNC |
| DEN | DEN | DEN |
| DATA | DATA[17:0] | DATA[15:0] |

5.7 Application Notes

5.7.1 DMA transfer mode

There are three DMA transfer modes for loading win1 frame data determined by following parameters(X=0,1):

dma_burst_length
winX_no_outstanding
winX_gather_en
winX_gather_thres

5.7.2 Auto outstanding transfer mode(random transfer)

When winX_no_outstanding is 0, multi-bursts transfer command could be sent out to AXI master interface continuously if the internal memory has enough space to store new data. The continuous random burst number is in the range of 1 to 4, mainly depending on the empty level of internal memory, dma_burst_length, data format and active image width.

1. Configured outstanding transfer mode(fixed transfer)

When winX_gather_en is 1, fixed-number of bursts transfer command should be sent out to AXI master interface continuously if the internal memory has enough space to store new data. The fixed-number is determined by winX_gather_thres. Since the internal memory size is limited, there is some restriction for the winX_gather_thres as follows.

Table 5-3 Gather configuration for all format

| Gather Threshold | dma_burst_length =2'b00(burst16) | dma_burst_length =2'b01(burst8) | dma_burst_length =2'b10(burst4) |
|------------------|----------------------------------|---------------------------------|---------------------------------|
| ARGB888 | | | |
| RGB888 | 0,1,2,3 | 0,1,2,3 | 0,1,2,3 |
| RGB565 | | | |
| 8BPP | 0,1,2,3 | 0,1,2,3 | 0,1,2,3 |

5.7.3 GAMMA LUT

When dsp_lut_en is 0, the DSP LUT data should be refreshed by software. i.e, writing dsp lut data to the internal memory with the start address DSP_LUT_MST. The memory size is 256x24, i.e, lower 24bits valid, and the writing data number is determined by software.

5.7.4 DMA control (QoS/outstanding)

If you want to get higher priority for VOP to access external memory when the frame data is urgent, a QoS and hurry request can be generated and sent out basing on the configured values:

sw_noc_qos_en: SYS_CTRL1[0]
sw_noc_qos_value: SYS_CTRL1[2:1]
sw_noc_hurry_en: SYS_CTRL1[4]
sw_noc_hurry_value: SYS_CTRL1[6:5]
sw_noc_hurry_threshold: SYS_CTRL1[11:8]
sw_axi_max_outstand_en: SYS_CTRL1[12]
sw_axi_max_outstand_num: SYS_CTRL1[20:16]

QoS request for higher bus priority for win1
max outstanding num is configurable.

5.7.5 Interrupt

VOP interrupt is comprised of 11 interrupt sources:

- dsp hold interrupt
- frame start0 interrupt
- frame start1 interrupt
- address same interrupt
- line flag0 interrupt

- line flag1 interrupt
- bus error interrupt
- win1 empty interrupt
- DMA finish interrupt

Every interrupt has independent interrupt enable signal(VOP_INT_EN),interrupt clear signal(VOP_INT_CLR) and interrupt raw status signal(VOP_INT_STATUS).

There is only one interrupt combined with all this interrupt signals to cpu, and it high active.

Dma finish interrupt is used for changing DDR frequency. This interrupt will be asserted when dma finish getting all the Pixel DATA every frame. This interrupt is asserted at the end of the frame.

Address same interrupt will be asserted at this kind of scenario that all the memory start address configured in VOP reg are same compared with the former frame. This interrupt is asserted at the beginning of frame start.

The difference between frame start0 interrupt and frame start1 interrupt is that frame start0 interrupt will be asserted every frame, while the frame start1 interrupt will be masked when all the memory start address are same.

5.7.6 RGB display mode

RGB display mode is used for RGB panel display. It is a continuous frames display mode.

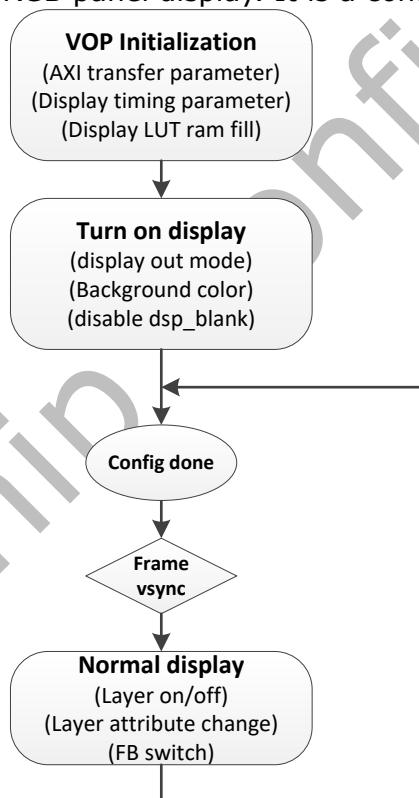


Fig. 5-15 VOP RGB Mode Programming Flow

1.VOP initialization

VOP initialization should be done before turning display on.

Firstly, AXI bus parameter (VOP_SYS_CTRL1) should be set for DMA transfer.

Secondly, display panel/interface timing should be set for display output. The registers are: VOP_DSP_HTOTAL_HS_END/ VOP_DSP_HACT_ST_END/ VOP_DSP_VTOTAL_HS_END/ VOP_DSP_VACT_ST_END

2.Background display

Before normal display, the background display could be turn on.

Firstly, set display output mode (VOP_DSP_CTRL0/1) according to display device.

Secondly, disable dsp_blank mode, which would not be enable until frame synchronization. Finally, writing '1' to "VOP_REG_CFG_DONE" register then all the frame-sync registers will

be enable at the beginning of next frame.

3.Normal display

In normal display, all the display layers' attribute could be different according display scenario. So there is a programming loop in this mode.

Firstly, configure all the display layers' attribute registers for the change of image format, location, size, scaling factor, alpha and overlay and so on. Those register would not be enable until frame synchronization.

Finally, write 1 to "VOP_REG_CFG_DONE" register then all the frame-sync registers will be enable at the beginning of next frame.

5.7.7 Immediately control register

There are two types of registers in VOP , one is effective immediately, the other is effective by frame sync. Effective immediately registers list as follows, other registers are all effective by frame sync.

Table 5-4 Effective immediately register table

| register address | description |
|------------------|--|
| 0x0008 | background |
| 0x0018 | All display and control signal registers |
| 0xe0~0xf4 | Frc configuration bits |

5.7.8 Output Polarity Control

RGB and MIPI are supported, each channel has its own xxx_dclk_en, xxx_dclk_pol, xxx_hsync_pol, xxx_vsync_pol, xxx_den_pol.

The xxx_dclk_en should be set to 1 when output select the xxx channel, and the other channel's xxx_dclk_en should be set to 0 to gate the output clk and data.

When using RGB panel, the dclk should be tied to "0" or "1" in some scenarios.

In this case, you should enable sw_io_pad_clk_sel, to tie dclk to "0". If enable rgb_dclk_pol at the same time, the dclk will tie to "1".

5.7.9 Some special control

- The blanking value of VSYNC could be configured through reg BLANKING_VALUE;
- The current scan line number could be read through reg_addr 0x190;
- There is a FLAG_REG which is readable and writable. This FLAG_REG does not for config function, our software staff may use it in the future. After writing a meaningful 32bit value to FLAG_REG, we can read it before frame valid through reading 0x1fc value, and can get the same value after frame valid reading 0x1f8 value;

5.7.10 RGB PATH

TTL IO typical configuration reference to GRF chapter.

Chapter 6 Video Output Processor (VOP_RAW)

6.1 Overview

Video Output Processor is a dma engine and a display interface from memory frame buffer to CSI DEVICE. VOP is connected to an AHB bus through an AHB slave and AXI bus through an AXI master. The register setting is configured through the AHB slave interface and the display frame data is read through the AXI master interface.

6.1.1 Features

VOP supports the following features:

- Clock
 - Hclk 200Mhz
 - Aclk 400Mhz
 - Dclk 300Mhz
- Display interface
 - Parallel PDAF Interface
 - Max output resolution: 5k(16M cam)
 - Support max timing 8k
- Layer process
 - Background layer
 - ◆ Programmable 10bit raw
 - Win layer
 - ◆ Support RAW8/RAW10
 - ◆ Support virtual display
 - ◆ Master address 64bit aligned
 - ◆ Stride 64bit aligned
- Bus interface
 - Support AMBA 2.0 AHB slave interface for accessing internal registers,32bit data bus width
 - Support AMBA 3.0 AXI master read interface for loading frame data,64bit data bus width
- interrupt
 - One combined interrupt
 - ◆ High active
 - ◆ Combinational interrupt sources
 - ✧ frame start interrupt
 - ✧ frame end interrupt
 - ✧ line flag0 interrupt
 - ✧ line flag1 interrupt
 - ✧ bus error interrupt
 - ✧ win empty interrupt
 - ✧ DMA finish interrupt

6.2 Function Description

6.2.1 Block Diagram

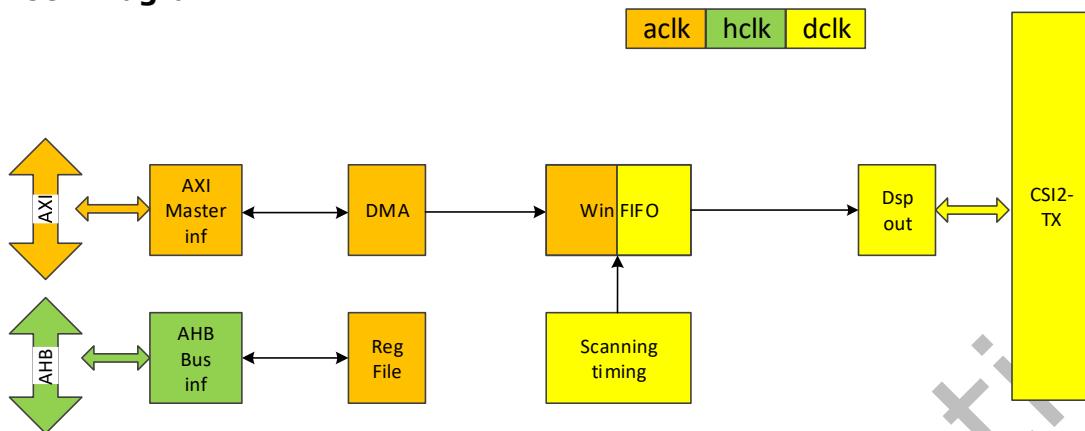


Fig. 6-1 VOP Block Diagram

6.2.2 Internal DMA

Internal DMA can fetch the pixel data through AXI bus from system memory (DDR) for all the display layers. Data fetching is driven by display output requirement.

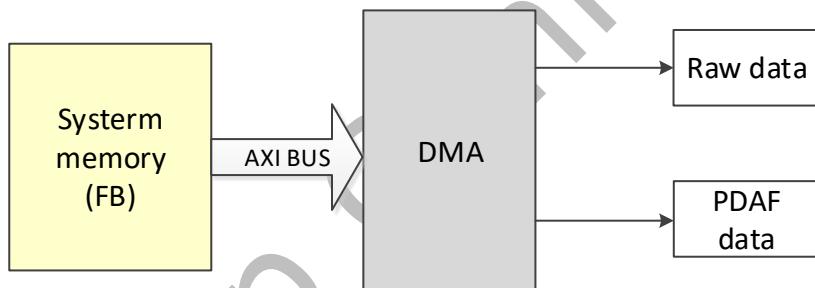


Fig. 6-2 VOP Internal DMA

6.2.3 Virtual display

Virtual display is supported in Win. The active image is part of the virtual (original) image in frame buffer memory. The virtual width is indicated by setting WIN_VIR_STRIDE for different data format.

The virtual stride should be multiples of word (64-bit). That means dummy bytes in the end of virtual line if the real pixels are not 64-bit aligned.

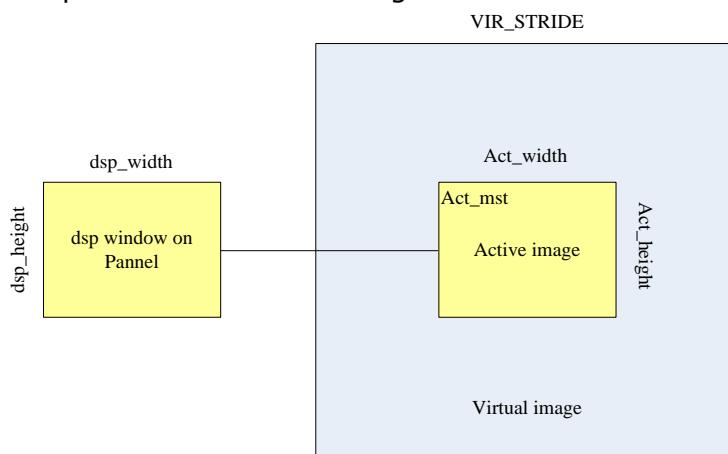


Fig. 6-3 VOP Virtual Display Mode

6.2.4 Hold mode and continuous mode

Hold mode, in hold mode need to configure the frame start to trigger a frame transmission, it will not scanning timing (into hold mode) after one frame transmission.
Continuous mode, in continuous mode, the vop is always scanning timing, and always transmits data.

6.2.5 Ping-pong mode

In ping-pong mode will have two raw addresses and two padf addresses, vop read data automatically switching between the two addresses.

6.2.6 PDAF mode

There are three different PDAF modes supported in the vop, vblank PDAF, hblank PDAF and hvblank mode.

In vblank mode, the PDAF data will transfer in vertical front porch or in vertical back porch or in both. The size of the PDAF is equal to the size of actual raw image size. User defined datatype is also allowed in the vblank mode, each vertical porch supports 4 different types at most.

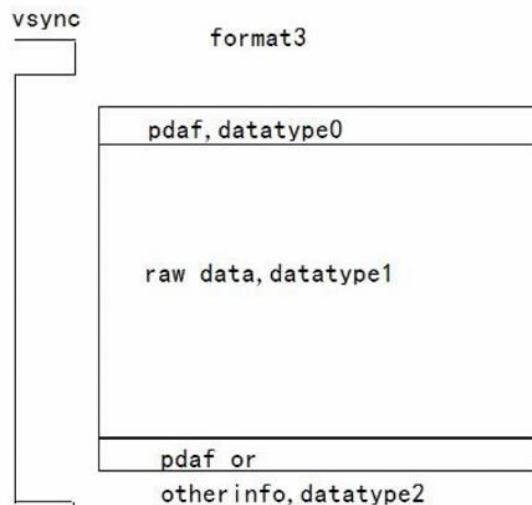


Fig. 6-4 VOP PDAF vblank mode

In hblank mode, the PDAF will transfer in horizontal front porch. The maximum size of the PDAF is 256 pixels.

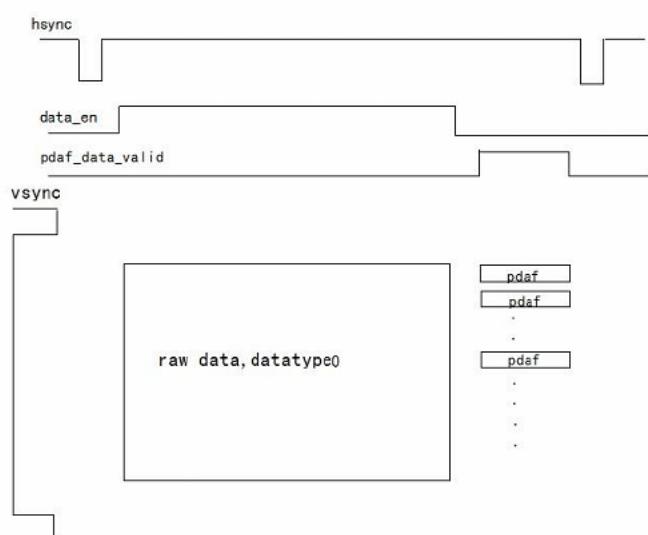


Fig. 6-5 VOP PDAF hblank mode

In the hvblank mode, the PDAF will transfer in both vertical porch and horizontal front porch. The type of the PDAF data in these three regions will be different.

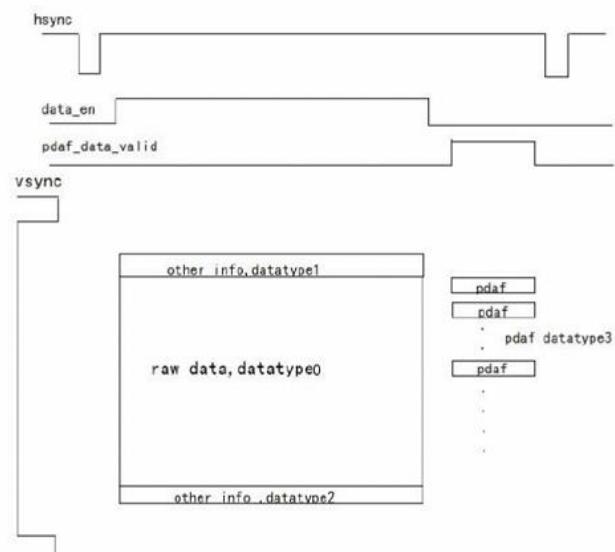


Fig. 6-5 VOP PDAF hvblank mode

6.2.7 Interweave mode

In hblank mode and hvblank mode, interweave hblank mode is supported besides normal hblank mode. In this mode, the PDAF data will interweave with raw data during the transmission by configurable PDAF hblank pattern.

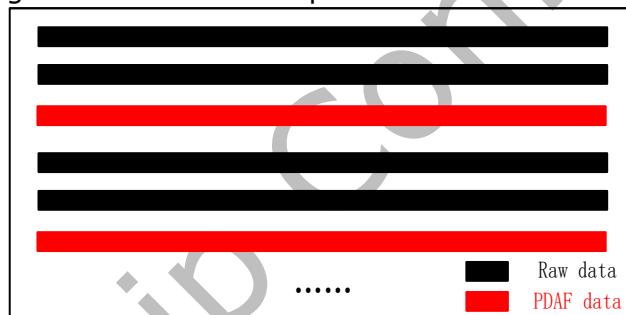


Fig. 6-6 VOP PDAF interweave mode

6.3 Timing Diagram

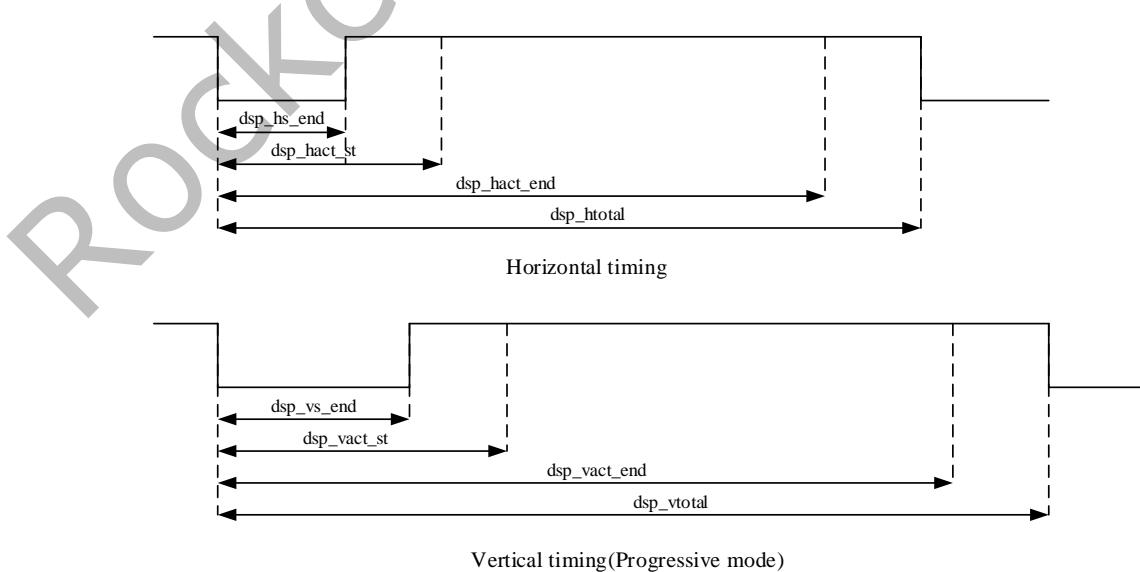


Fig. 6-7 VOP camera interface timing setting

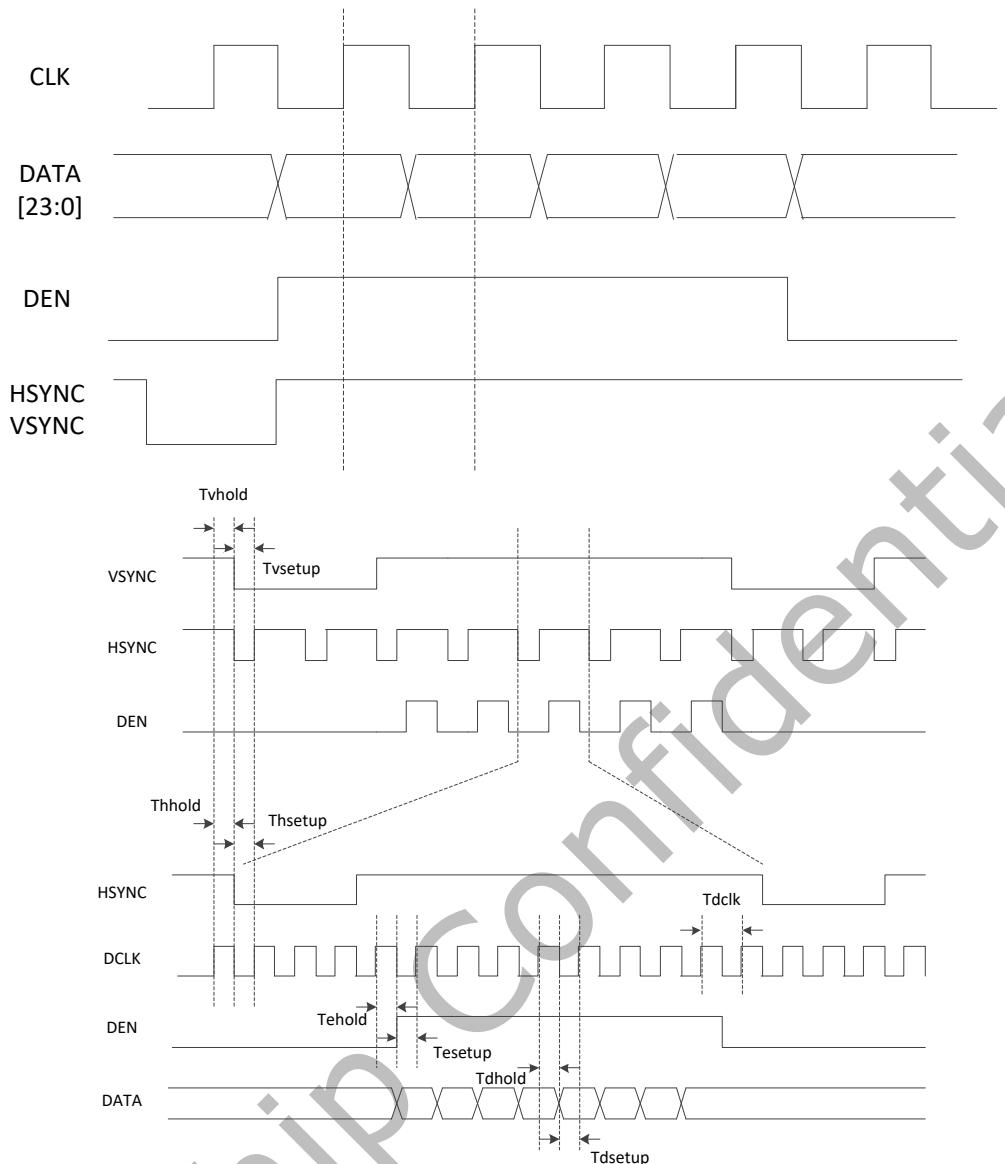


Fig. 6-8 VOP RGB Interface Timing(SDR)

6.4 Register Description

6.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

6.4.2 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|----------------------|--------|------|-------------|---------------------------|
| VOP_RAW_REG_CFG_DONE | 0x0000 | W | 0x00000000 | Register config done flag |
| VOP_RAW_VERSION | 0x0004 | W | 0x00000000 | Version for vop |
| VOP_RAW_DSP_BG | 0x0008 | W | 0x00000000 | Dsp background register |
| VOP_RAW_LINE_FLAG | 0x000c | W | 0x00000000 | Line flag register |
| VOP_RAW_SYS_CTRL0 | 0x0010 | W | 0x00000002 | System control register0 |
| VOP_RAW_SYS_CTRL1 | 0x0014 | W | 0x00000000 | System control register1 |

| Name | Offset | Size | Reset Value | Description |
|------------------------------|--------|------|-------------|--|
| VOP_RAW_DSP_CTRL0 | 0x0018 | W | 0x00000000 | Dsp control register |
| VOP_RAW_DSP_DBG0 | 0x001c | W | 0x00000000 | Debug register |
| VOP_RAW_PDAF_CTRL | 0x0020 | W | 0x00000000 | PDAF control register |
| VOP_RAW_PDAF_HBLANK_CTRL | 0x0024 | W | 0x00000000 | PDAF control register |
| VOP_RAW_PDAF_MST0 | 0x0028 | W | 0x00000000 | PDAF memory start address0 (hblank) |
| VOP_RAW_PDAF_MST1 | 0x002c | W | 0x00000000 | PDAF memory start address1 (hblank) |
| VOP_RAW_INTR_EN | 0x0030 | W | 0x00000000 | Interrupt enable register |
| VOP_RAW_INTR_CLEAR | 0x0034 | W | 0x00000000 | Interrupt clear register |
| VOP_RAW_INTR_STATUS | 0x0038 | W | 0x00000000 | Interrupt status register |
| VOP_RAW_INTR_RAW_STATUS | 0x003c | W | 0x00000000 | Interrupt raw status register |
| VOP_RAW_WIN1_CTRL0 | 0x0040 | W | 0x00000000 | Win1 ctrl register0 |
| VOP_RAW_WIN1_CTRL1 | 0x0044 | W | 0x00006580 | Win1 ctrl register0 |
| VOP_RAW_WIN1_VIR | 0x0048 | W | 0x00000000 | Win1 virtual stride |
| VOP_RAW_PDAF_DSP_IN_FO | 0x004c | W | 0x00ef013f | PDAF display width/height on panel (hblank mode) |
| VOP_RAW_WIN1_MST | 0x0050 | W | 0x00000000 | Win1 memory start address |
| VOP_RAW_WIN1_DSP_IN_FO | 0x0054 | W | 0x00ef013f | Win1 display width/height on panel |
| VOP_RAW_WIN1_DSP_ST | 0x0058 | W | 0x000a000a | Win1 display start point on panel |
| VOP_RAW_WIN1_MST1 | 0x005c | W | 0x00000000 | Win1 memory start address1 in ping-pong mode |
| VOP_RAW_DSP_HTOTAL_HS_END | 0x0060 | W | 0x015a0008 | Panel scanning horizontal width and hsync pulse end point |
| VOP_RAW_DSP_HACT_ST_END | 0x0064 | W | 0x000a014a | Panel active horizontal scanning start point and end point |
| VOP_RAW_DSP_VTOTAL_VS_END | 0x0068 | W | 0x01200006 | Panel scanning vertical height and vsync pulse end point |
| VOP_RAW_DSP_VACT_ST_END | 0x006c | W | 0x000f00ff | Panel active vertical scanning start point and end point |
| VOP_RAW_DSP_PDAF_HACT_ST_END | 0x0070 | W | 0x014c0154 | PDAF active start point and end point |
| VOP_RAW_DSP_EX0_VACT_ST_END | 0x0074 | W | 0x00080009 | PDAF extra0 vertical active start point and end point |
| VOP_RAW_DSP_EX1_VACT_ST_END | 0x0078 | W | 0x0009000a | PDAF extra1 vertical active start point and end point |
| VOP_RAW_DSP_EX2_VACT_ST_END | 0x007c | W | 0x000a000b | PDAF extra2 vertical active start point and end point |
| VOP_RAW_DSP_EX3_VACT_ST_END | 0x0080 | W | 0x000b000c | PDAF extra3 vertical active start point and end point |

| Name | Offset | Size | Reset Value | Description |
|--|--------|------|-------------|---|
| VOP_RAW_DSP_EX4_VAC_T_ST_END | 0x0084 | W | 0x01000101 | PDAF extra4 vertical active start point and end point |
| VOP_RAW_DSP_EX5_VAC_T_ST_END | 0x0088 | W | 0x01010102 | PDAF extra5 vertical active start point and end point |
| VOP_RAW_DSP_EX6_VAC_T_ST_END | 0x008c | W | 0x01020103 | PDAF extra6 vertical active start point and end point |
| VOP_RAW_DSP_EX7_VAC_T_ST_END | 0x0090 | W | 0x01030104 | PDAF extra7 vertical active start point and end point |
| VOP_RAW_DSP_PDAF_INTERWEAVE_VACT_ST_EN_D | 0x0094 | W | 0x000f00ff | PDAF vertical active start point and end point in interweave mode |
| VOP_RAW_PDAF_RREQ_PRIORITY_VALUE | 0x0098 | W | 0x00000000 | PDAF read request priority value |
| VOP_RAW_PDAF_EX_VIR | 0x009c | W | 0x00000000 | PDAF extra virtual stride |
| VOP_RAW_PDAF_EX_FRT_MST | 0x00a0 | W | 0x00000000 | Front PDAF extra master address0 |
| VOP_RAW_PDAF_EX_FRT_MST1 | 0x00a4 | W | 0x00000000 | Front PDAF extra master address1 |
| VOP_RAW_PDAF_EX_END_MST | 0x00a8 | W | 0x00000000 | Rear PDAF extra master address0 |
| VOP_RAW_PDAF_EX_END_MST1 | 0x00ac | W | 0x00000000 | Rear PDAF extra master address1 |
| VOP_RAW_PDAF_EX_WC0 | 0x00b0 | W | 0x00000000 | PDAF ex type word count0 |
| VOP_RAW_PDAF_EX_WC1 | 0x00b4 | W | 0x00000000 | PDAF ex type word count1 |
| VOP_RAW_PDAF_EX_WC2 | 0x00b8 | W | 0x00000000 | PDAF ex type word count2 |
| VOP_RAW_PDAF_EX_WC3 | 0x00bc | W | 0x00000000 | PDAF ex type word count3 |
| VOP_RAW_PDAF_EX_TYP_E_CTRL0 | 0x00c0 | W | 0x00000000 | Extra PDAF type control register0 |
| VOP_RAW_PDAF_EX_TYP_E_CTRL1 | 0x00c4 | W | 0x00000000 | Extra PDAF type control register1 |
| VOP_RAW_PDAF_EX_EN_CTRL | 0x00c8 | W | 0x00000000 | Extra PDAF control register |

Notes: **S**-Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

6.4.3 Detail Register Description

VOP_RAW_REG_CFG_DONE

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | Reserved |
| 0 | WO | 0x0 | reg_load_global_en In the first setting of the register, the new value was saved into the mirror register. When all the register configured finish, writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame. |

VOP_RAW_VERSION

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | major used for IP structure |
| 23:16 | RO | 0x00 | minor big feature change under same structure |
| 15:0 | RO | 0x0000 | build rtl current svn number |

VOP_RAW_DSP_BG

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------------------|
| 31:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0x00 | dsp_bg_raw Background raw data |

VOP_RAW_LINE_FLAG

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | Reserved |
| 27:16 | RW | 0x000 | dsp_line_flag1_num The display line number when the flag interrupt 1 occur, the range is (0~ DSP_VTOTAL-1). |
| 15:12 | RO | 0x0 | Reserved |
| 11:0 | RW | 0x000 | dsp_line_flag0_num The display line number when the flag interrupt occur, the range is (0~ DSP_VTOTAL-1). |

VOP_RAW_SYS_CTRL0

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | Reserved |
| 15 | RW | 0x0 | auto_gating_en auto gating enable |
| 14:4 | RO | 0x0 | Reserved |
| 3 | RW | 0x0 | bypass_en 1'b0: Disable bypass 1'b1: Enable bypass |
| 2 | RW | 0x0 | Pingpong_mode If in normal continuous display mode, there two mode as follow. 1'b0: Normal mode 1'b1: Ping-pong mode |
| 1 | RW | 0x1 | hold_mode when hold mode, configure fs_hold_mode to start one frame display, then into hold mode . If in normal continuo mode, configure hold mode, it will into hold when frame end. 1'b0: Normal continuous display mode 1'b1: Hold mode, one frame display |
| 0 | W1 C | 0x0 | frame_st frame start trigger signal , write 1 then auto clear 1'b1: Start |

VOP_RAW_SYS_CTRL1

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | Reserved |
| 15:12 | RW | 0x0 | imd_noc_hurry_threshold the noc hurry trigger threshold value,0~15 |
| 11:10 | RW | 0x0 | imd_noc_hurry_value noc hurry value,0~3 |
| 9 | RW | 0x0 | imd_noc_hurry_en 1'b0: Disable 1'b1: Enable |
| 8:4 | RW | 0x00 | sw_axi_max_outstand_num the max outstanding number,0~31. |
| 3 | RW | 0x0 | sw_axi_max_outstand_en 1'b0: Disable 1'b1: Enable |
| 2:1 | RW | 0x0 | sw_noc_qos_value noc qos level,0~3 |
| 0 | RW | 0x0 | sw_noc_qos_en 1'b0: Disable 1'b1: Enable |

VOP_RAW_DSP_CTRL0

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | Reserved |
| 5 | RW | 0x0 | PDAF_den_pol 1'b0: Positive 1'b1: Negative |
| 4 | RW | 0x0 | MIPI_den_pol 1'b0: Positive 1'b1: Negative |
| 3 | RW | 0x0 | MIPI_vsync_pol 1'b0: Negative 1'b1: Positive |
| 2 | RW | 0x0 | MIPI_hsync_pol 1'b0: Negative 1'b1: Positive |
| 1 | RW | 0x0 | MIPI_dclk_pol 1'b0: MIPI dclk inv disable 1'b1: MIPI dclk inv enable |
| 0 | RW | 0x0 | MIPI_dclk_en 1'b0: MIPI dclk disable 1'b1: MIPI dclk enable |

VOP_RAW_DSP_DBG0

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | PDAF_wc PDAF word counter (hblank) |
| 15:12 | RO | 0x0 | Reserved |
| 11:0 | RO | 0x000 | scan_line_num the scanning line number |

VOP_RAW_PDAF_CTRL

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:14 | RO | 0x0 | Reserved |
| 13:8 | RW | 0x00 | PDAF_hblank_data_type The PDAF_hblank_data_type is the only datatype in hblank mode. |
| 7:6 | RW | 0x0 | PDAF_valid_bits 2'b00: 8 bits valid 2'b01: 10 bits valid 2'b10: 16 bits valid 2'b11: Reserved |
| 5:4 | RW | 0x0 | PDAF_vc_num define the virtual ID to be used by CSI TX. |
| 3 | RW | 0x0 | PDAF_hblank_interweave_en 1'b0: Disable 1'b1: Enable |
| 2:1 | RW | 0x0 | PDAF_type 2'bx1: Hblakk type 2'b1x: Vblank type |
| 0 | RW | 0x0 | PDAF_en 1'b0: PDAF disable 1'b1: PDAF enable |

VOP RAW PDAF HBLANK CTRL

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0000 | dsp_PDAF_hblank_mask_height mask_height in a PDAF hblank counter cycle (if 0 PDAF hblank hden always disable) if PDAF_hblank_counter > unmask_height-1 and <= unmask_height +mask_height-1,PDAF hblank hden enable |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0000 | dsp_PDAF_hblank_unmask_height unmask_height in a PDAF hblank counter cycle (if 0 and mask_height not 0 then then PDAF hblank hden always enable) if PDAF_hblank_counter <= unmask_height-1 then PDAF hblank hden disable |

VOP RAW PDAF MST

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | PDAF_mst 32 bits master address(hblank) |

VOP RAW PDAF MST1

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | PDAF_mst1 32 bits master address1(hblank) |

VOP_RAW_INTR_EN

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | write_mask When every bit HIGH, enable the writing corresponding bit When every bit LOW, don't care the writing corresponding bit |
| 15:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | dma_frm_fsh_intr_en 1'b0: Disable 1'b1: Enable |
| 6 | RW | 0x0 | win1_empty_intr_en 1'b0: Disable 1'b1: Enable |
| 5 | RW | 0x0 | line_flag1_intr_en 1'b0: Disable 1'b1: Enable |
| 4 | RW | 0x0 | line_flag0_intr_en 1'b0: Disable 1'b1: Enable |
| 3 | RW | 0x0 | bus_error_intr_en 1'b0: Disable 1'b1: Enable |
| 2 | RW | 0x0 | fe_intr_en 1'b0: Disable 1'b1: Enable |
| 1 | RW | 0x0 | fs_intr_en 1'b0: Disable 1'b1: Enable |
| 0 | RW | 0x0 | hold_intr_en dsp hold interrupt enable |

VOP_RAW_INTR_CLEAR

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7 | W1 C | 0x0 | dma_frm_fsh_intr_clr dma frame finish interrupt clear(Auto clear) |
| 6 | W1 C | 0x0 | win0_empty_intr_clr win0 data empty interrupt clear(Auto clear) |
| 5 | W1 C | 0x0 | line_flag1_intr_clr Line flag 1 Interrupt clear(Auto clear) |
| 4 | W1 C | 0x0 | line_flag0_intr_clr Line flag 0 Interrupt clear(Auto clear) |
| 3 | W1 C | 0x0 | bus_error_intr_clr Bus error Interrupt clear(Auto clear) |
| 2 | W1 C | 0x0 | fe_intr_clr Frame end interrupt clear (Auto clear) |
| 1 | W1 C | 0x0 | fs_intr_clr Frame start interrupt clear (Auto clear) |
| 0 | W1 C | 0x0 | hold_intr_clr hold interrupt clear (Auto clear) |

VOP_RAW_INTR_STATUS

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RO | 0x0 | dma_frm_fsh_intr_sts dma frame finish interrupt status |
| 6 | RO | 0x0 | win1_empty_intr_sts win1 data empty interrupt status |
| 5 | RO | 0x0 | line_flag1_intr_sts Line flag 1 Interrupt status |
| 4 | RO | 0x0 | line_flag0_intr_sts Line flag 0 Interrupt status |
| 3 | RO | 0x0 | bus_error_intr_sts Bus error Interrupt status |
| 2 | RO | 0x0 | fe_intr_sts Frame end interrupt status |
| 1 | RO | 0x0 | fs_intr_sts Frame start interrupt status |
| 0 | RO | 0x0 | hold_intr_sts Hold interrupt status |

VOP_RAW_INTR_RAW_STATUS

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RO | 0x0 | dma_frm_fsh_intr_raw_sts dma frame finish interrupt raw status |
| 6 | RO | 0x0 | win1_empty_intr_raw_sts win1 data empty interrupt raw status |
| 5 | RO | 0x0 | line_flag1_intr_raw_sts Line flag 1 Interrupt raw status |
| 4 | RO | 0x0 | line_flag0_intr_raw_sts Line flag 0 Interrupt raw status |
| 3 | RO | 0x0 | bus_error_intr_raw_sts Bus error Interrupt raw status |
| 2 | RO | 0x0 | fe_intr_raw_sts Frame end raw interrupt status |
| 1 | RO | 0x0 | fs_intr_raw_sts Frame start raw interrupt status |
| 0 | RO | 0x0 | hold_intr_raw_sts hold raw interrupt status |

VOP_RAW_WIN1_CTRL0

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | Reserved |
| 4 | RW | 0x0 | win1_no_outstanding win 1 no outstanding number |
| 3 | RW | 0x0 | win1_data_aligned valid when win_data_fmt == 2'b11, 1'b0: 10bit valid data low aligned ,valid data[9:0] 1'b1: 10bit valid data high aligned,valid data[15:6] |
| 2:1 | RW | 0x0 | win1_data_fmt 2'b00: Raw8 2'b01: Raw10 2'b10: Reserved 2'b11: Raw16(10bit valid) |
| 0 | RW | 0x0 | win1_en 1'b0: Win1 layer disable 1'b1: Win1 layer enable |

VOP_RAW_WIN1_CTRL1

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | Reserved |
| 15:12 | RW | 0x6 | sw_PDAF_rid axi read id of win1 channel |
| 11:8 | RW | 0x5 | sw_win1_rid axi read id of win1 channel |
| 7:4 | RW | 0x8 | win1_axi_gather_num axi gather number ,0~15. |
| 3:2 | RW | 0x0 | win1_dma_burst_length 2'b00: Burst16 (burst 15 in rgb888 pack mode) 2'b01: Burst8 (burst 12 in rgb888 pack mode) 2'b10: Burst4 (burst 6 in rgb888 pack mode) |
| 1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | win1_axi_gather_en 1'b0: Disable 1'b1: Enable |

VOP RAW WIN1 VIR

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:29 | RO | 0x0 | Reserved |
| 28:16 | RW | 0x0000 | PDAF_vir_stride Number of words of PDAF Virtual width (hblank) |
| 15:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x0000 | win1_vir_stride Number of words of Win1 Virtual width (including vblank except type) raw8: Win1_vir_width >> 2 raw10: Win1_vir_width * 5 / 4 (64bit aligned) raw8_10: Win1_vir_width >> 1 |

VOP RAW PDAF DSP INFO

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:29 | RO | 0x0 | Reserved |
| 28:16 | RW | 0x00ef | dsp_PDAF_height PDAF_dsp_height = (PDAF dsp vertical size -1) |
| 15:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x013f | dsp_PDAF_width PDAF_dsp_width = (PDAF dsp horizontal size -1) |

VOP RAW WIN1 MST

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | win1_mst Win1 frame buffer memory start address |

VOP RAW WIN1 DSP INFO

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:29 | RO | 0x0 | Reserved |
| 28:16 | RW | 0x00ef | dsp_win1_height win1_dsp_height = (win1 dsp vertical size -1) |
| 15:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x013f | dsp_win1_width win1_dsp_width = (win1 dsp horizontal size -1) |

VOP RAW WIN1 DSP ST

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x000a | dsp_win1_yst Win1 vertical start point(y) of the Panel scanning |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x000a | dsp_win1_xst Win1 horizontal start point(x) of the Panel scanning |

VOP RAW WIN1 MST1

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | win1_mst Win1 frame buffer memory start address |

VOP RAW DSP HTOTAL HS END

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x015a | dsp_htotal display scanning horizontal period |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0008 | dsp_hs_end display scanning hsync pulse width |

VOP RAW DSP HACT ST END

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x000a | dsp_hact_st display scanning horizontal active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x014a | dsp_hact_end display scanning horizontal active end point |

VOP_RAW_DSP_VTOTAL_VS_END

Address: Operational Base + offset (0x0068)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0120 | dsp_vtotal display scanning vertical period |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0006 | dsp_vs_end display scanning vsync pulse width (PDAF hblank line numbers should be extra added in interweave mode) |

VOP_RAW_DSP_VACT_ST_END

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x000f | dsp_vact_st display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x00ff | dsp_vact_end display scanning vertical active end point |

VOP_RAW_DSP_PDAF_HACT_ST_END

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x014c | PDAF_dsp_hact_st PDAF display scanning horizontal active start point (hblank) |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0154 | PDAF_dsp_hact_end PDAF display scanning horizontal active end point (hblank) |

VOP_RAW_DSP_EX0_VACT_ST_END

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0008 | ex0_dsp_vact_st ex0 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0009 | ex0_dsp_vact_end ex0 display scanning vertical active end point |

VOP RAW DSP EX1 VACT ST END

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0009 | ex1_dsp_vact_st ex1 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x000a | ex1_dsp_vact_end ex1 display scanning vertical active end point |

VOP RAW DSP EX2 VACT ST END

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x000a | ex2_dsp_vact_st ex2 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x000b | ex2_dsp_vact_end ex2 display scanning vertical active end point |

VOP RAW DSP EX3 VACT ST END

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x000b | ex3_dsp_vact_st ex3 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x000c | ex3_dsp_vact_end ex3 display scanning vertical active end point |

VOP RAW DSP EX4 VACT ST END

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0100 | ex4_dsp_vact_st ex4 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0101 | ex4_dsp_vact_end ex4 display scanning vertical active end point |

VOP RAW DSP EX5 VACT ST END

Address: Operational Base + offset (0x0088)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0101 | ex5_dsp_vact_st ex5 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0102 | ex5_dsp_vact_end ex5 display scanning vertical active end point |

VOP RAW DSP EX6 VACT ST END

Address: Operational Base + offset (0x008c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0102 | ex6_dsp_vact_st ex6 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0103 | ex6_dsp_vact_end ex6 display scanning vertical active end point |

VOP RAW DSP EX7 VACT ST END

Address: Operational Base + offset (0x0090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x0103 | ex7_dsp_vact_st ex7 display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0104 | ex7_dsp_vact_end ex7 display scanning vertical active end point (ex7_dsp_vact_end must be configued not less then dsp_vact_end no matter PDAF_ex7_en is enable or not , for generating the frame end interrupt) |

VOP RAW DSP PDAF INTERWEAVE VACT ST END

Address: Operational Base + offset (0x0094)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29:16 | RW | 0x000f | dsp_PDAF_interweave_vact_st display scanning vertical active start point |
| 15:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x00ff | dsp_PDAF_interweave_vact_end display scanning vertical active end point |

VOP RAW PDAF RREQ PRIORITY VALUE

Address: Operational Base + offset (0x0098)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | Reserved |
| 9:0 | RW | 0x000 | PDAF_fifo_wcnt_threshold_value PDAF fifo wcnt<=threshold value, change high rreq priority to win1 |

VOP RAW PDAF EX VIR

Address: Operational Base + offset (0x009c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x0000 | PDAF_ex_vir_stride Number of words of PDAF Virtual width (vblank) |

VOP RAW PDAF EX FRT MST

Address: Operational Base + offset (0x00a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | PDAF_ex_frt_mst 32 bits master address |

VOP RAW PDAF EX FRT MST1

Address: Operational Base + offset (0x00a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | PDAF_ex_frt_mst1 32 bits master address |

VOP RAW PDAF EX END MST

Address: Operational Base + offset (0x00a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | PDAF_ex_end_mst 32 bits master address |

VOP RAW PDAF EX END MST1

Address: Operational Base + offset (0x00ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | PDAF_ex_end_mst1 32 bits master address |

VOP RAW PDAF EX WC0

Address: Operational Base + offset (0x00b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | PDAF_ex1_wc PDAF ex1 type word counter |
| 15:0 | RW | 0x0000 | PDAF_ex0_wc PDAF ex0 type word counter |

VOP RAW PDAF EX WC1

Address: Operational Base + offset (0x00b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | PDAF_ex3_wc PDAF ex3 type word counter |
| 15:0 | RW | 0x0000 | PDAF_ex2_wc PDAF ex2 type word counter |

VOP RAW PDAF EX WC2

Address: Operational Base + offset (0x00b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | PDAF_ex5_wc PDAF ex5 type word counter |
| 15:0 | RW | 0x0000 | PDAF_ex4_wc PDAF ex4 type word counter |

VOP RAW PDAF EX WC3

Address: Operational Base + offset (0x00bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | PDAF_ex7_wc PDAF ex7 type word counter |
| 15:0 | RW | 0x0000 | PDAF_ex6_wc PDAF ex6 type word counter |

VOP RAW PDAF EX TYPE CTRL0

Address: Operational Base + offset (0x00c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:30 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 29:24 | RW | 0x00 | ex3_data_type ex3 data type in vblank mode. |
| 23:22 | RO | 0x0 | Reserved |
| 21:16 | RW | 0x00 | ex2_data_type ex2 data type in vblank mode. |
| 15:14 | RO | 0x0 | Reserved |
| 13:8 | RW | 0x00 | ex1_data_type ex1 data type in vblank mode. |
| 7:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x00 | ex0_data_type ex0 data type in vblank mode. |

VOP RAW PDAF EX TYPE CTRL1

Address: Operational Base + offset (0x00c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:24 | RW | 0x00 | ex7_data_type ex7 data type in vblank mode. |
| 23:22 | RO | 0x0 | Reserved |
| 21:16 | RW | 0x00 | ex6_data_type ex6 data type in vblank mode. |
| 15:14 | RO | 0x0 | Reserved |
| 13:8 | RW | 0x00 | ex5_data_type ex5 data type in vblank mode. |
| 7:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x00 | ex4_data_type ex4 data type in vblank mode. |

VOP RAW PDAF EX EN CTRL

Address: Operational Base + offset (0x00c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | PDAF_ex7_en 1'b0: Ex7 disable 1'b1: Ex7 enable |
| 6 | RW | 0x0 | PDAF_ex6_en 1'b0: Ex6 disable 1'b1: Ex6 enable |
| 5 | RW | 0x0 | PDAF_ex5_en 1'b0: Ex5 disable 1'b1: Ex5 enable |
| 4 | RW | 0x0 | PDAF_ex4_en 1'b0: Ex4 disable 1'b1: Ex4 enable |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 3 | RW | 0x0 | PDAF_ex3_en 1'b0: Ex3 disable 1'b1: Ex3 enable |
| 2 | RW | 0x0 | PDAF_ex2_en 1'b0: Ex2 disable 1'b1: Ex2 enable |
| 1 | RW | 0x0 | PDAF_ex1_en 1'b0: Ex1 disable 1'b1: Ex1 enable |
| 0 | RW | 0x0 | PDAF_ex0_en 1'b0: Ex0 disable 1'b1: Ex0 enable |

Chapter 7 Raster Graphic Acceleration(RGA)

7.1 Overview

RGA is a separate 2D raster graphic acceleration unit. It accelerates 2D graphics operations, such as image scaling, rotation, BitBLT, alpha blending and fading. RGA supports the following features:

- **Data format**
 - Input data:
 - ◆ Support ARGB/RGB888/RGB565/RGB4444/RGB5551/YUV420/YUV422
 - ◆ Support YUV422SP10bit/YUV420SP10bit(YUV-8bits out)
 - Output data:
 - ◆ ARGB/RGB888/RGB565/RGB4444/RGB5551/YUV420/YUV422
 - Pixel Format conversion, BT.601/BT.709
 - Dither operation
 - Max resolution: 8192x8192 source, 4096x4096 destination
- **Scaling**
 - Down-scaling: Average filter
 - Up-scaling: Bi-cubic filter(source>2048 would use Bi-linear)
 - Arbitrary non-integer scaling ratio, from 1/8 to 8
- **Rotation**
 - 0, 90, 180, 270 degree rotation
 - x-mirror, y-mirror& rotation operation
- **BitBLT**
 - Block transfer
 - Color palette/Color fill, support with alpha
 - Transparency mode (color keying/stencil test, specified value/value range)
 - Two source BitBLT:
 - A+B=B only BitBLT, A support rotate&scale when B fixed
 - A+B=C second source (B) has same attribute with (C) plus rotation function
- **Alpha Blending**
 - New comprehensive per-pixel alpha(color/alpha channel separately)
 - Fading
 - SRC1(R2Y)&&SRC0(YUV)—alpha->DST(YUV)
- **MMU**
 - 4k/64k page size
 - Four channel: SRC/SRC1/DST/CMD, individual base address and enable control bit
 - TLB pre-fetch

7.2 Block Diagram

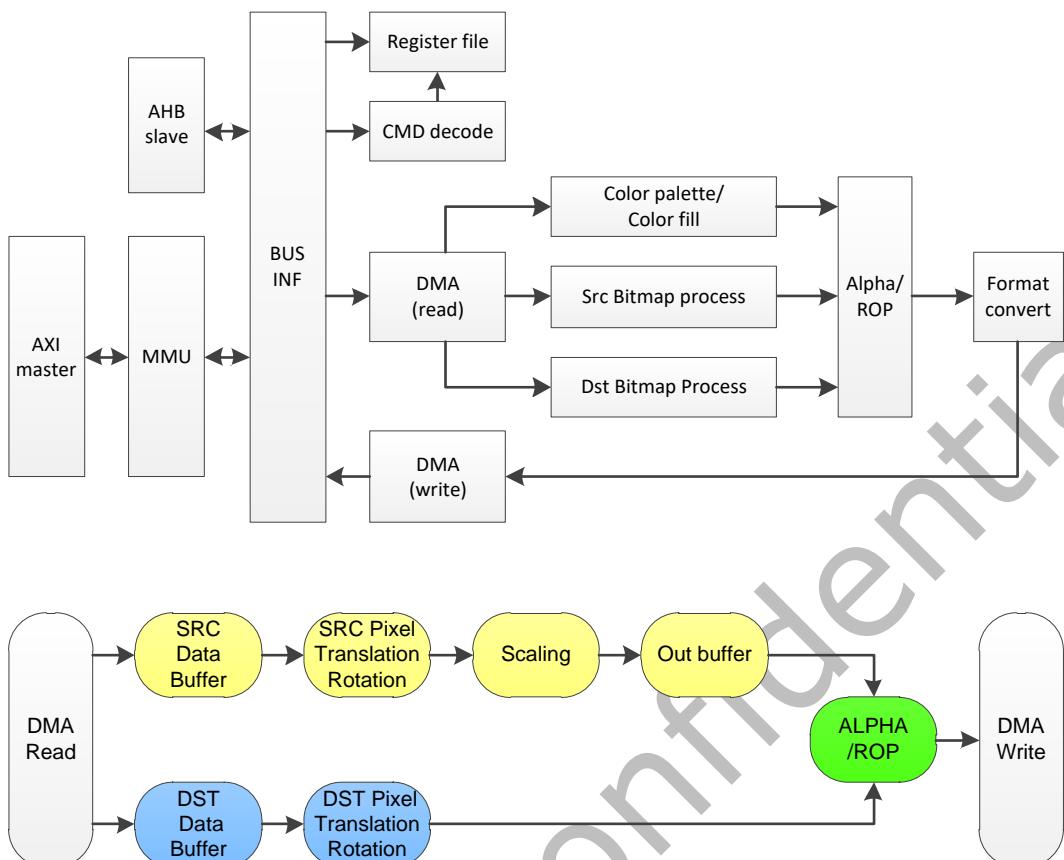


Fig. 7-1 RGA Block Diagram

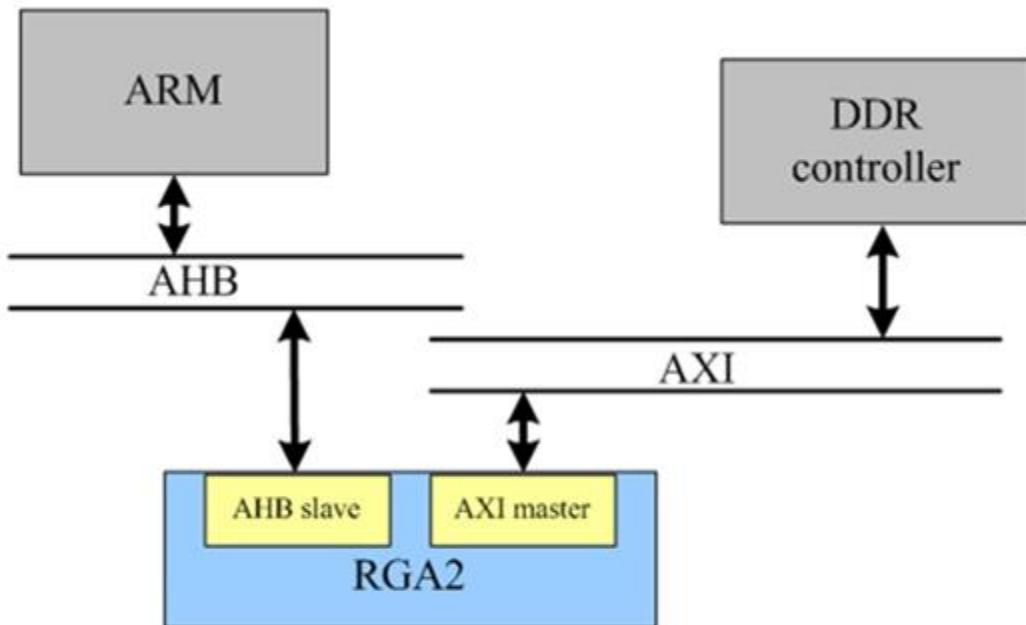


Fig. 7-2 RGA2 in SOC

7.3 Function Description

7.3.1 Data Format

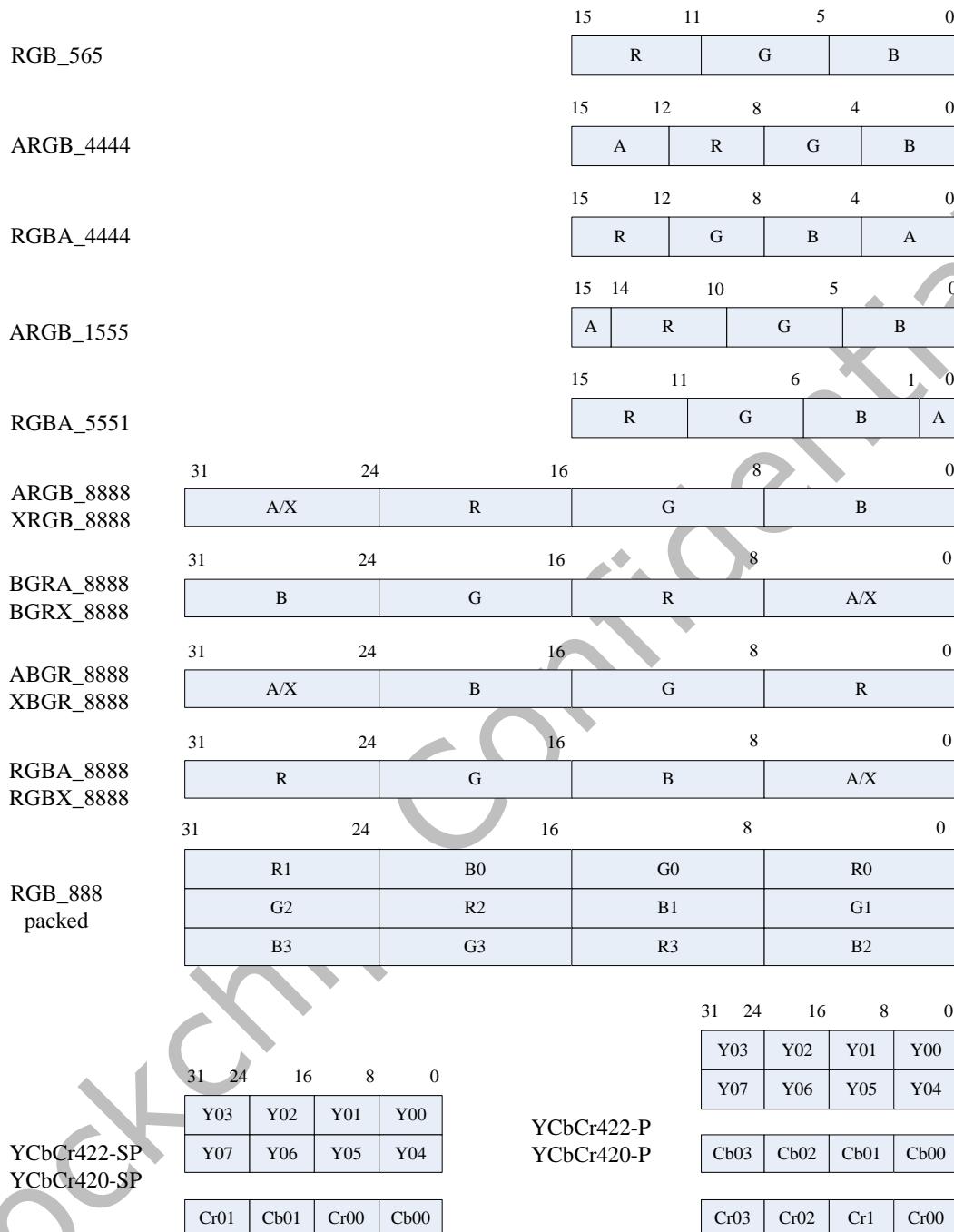


Fig. 7-3 RGA Input Data Format

All input datas (defined by SRC_IN_FMT/DST_IN_FMT) are converted to ABGR8888. The results are converted to the output data format (defined by DST_OUT_FMT).

7.3.2 Dithering

There could have dithering operation for source image when the source image format is not RGB565 and the destination format is RGB565.

The down-dithering is done using Dither Allegro.

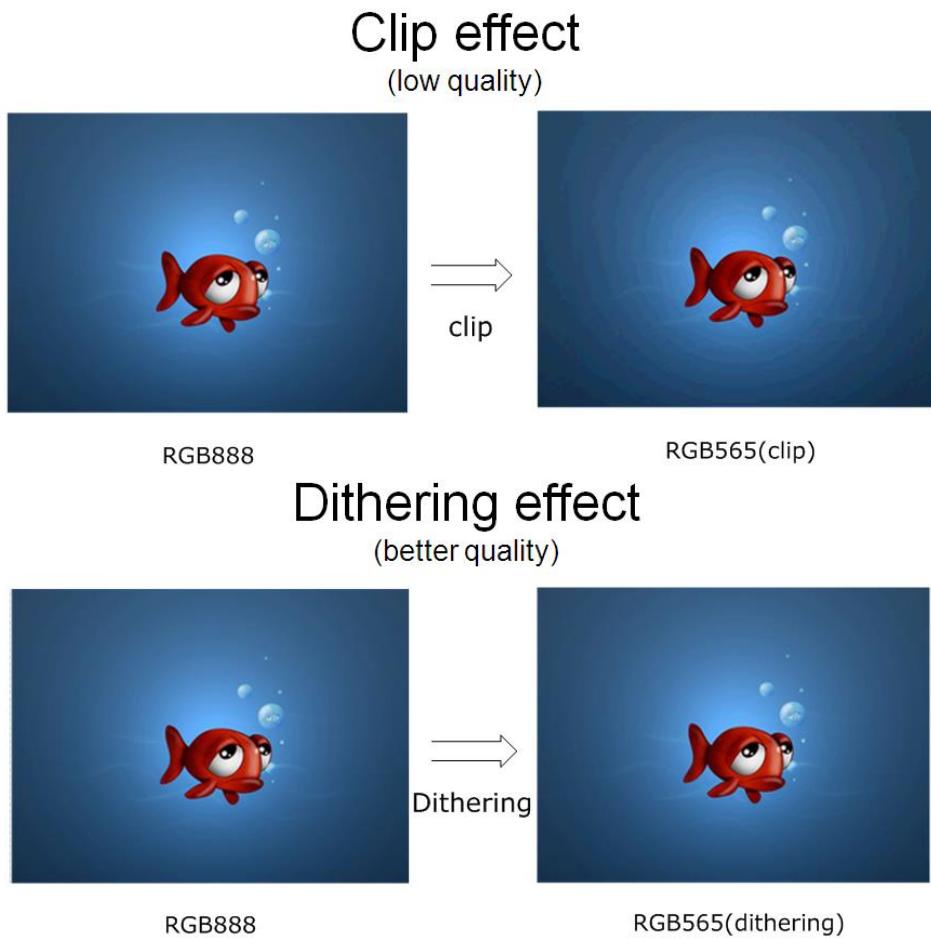


Fig. 7-4 RGA Dither effect

7.3.3 Alpha mode

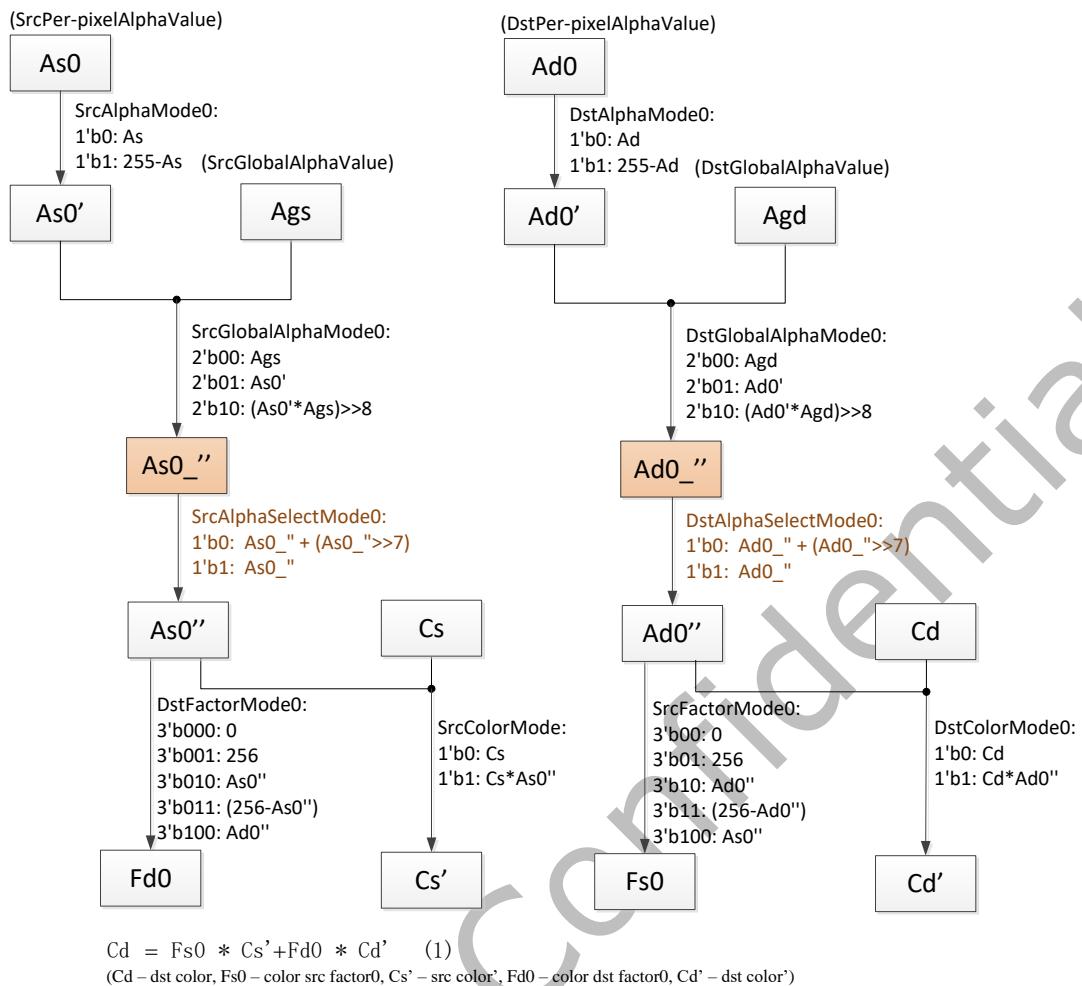
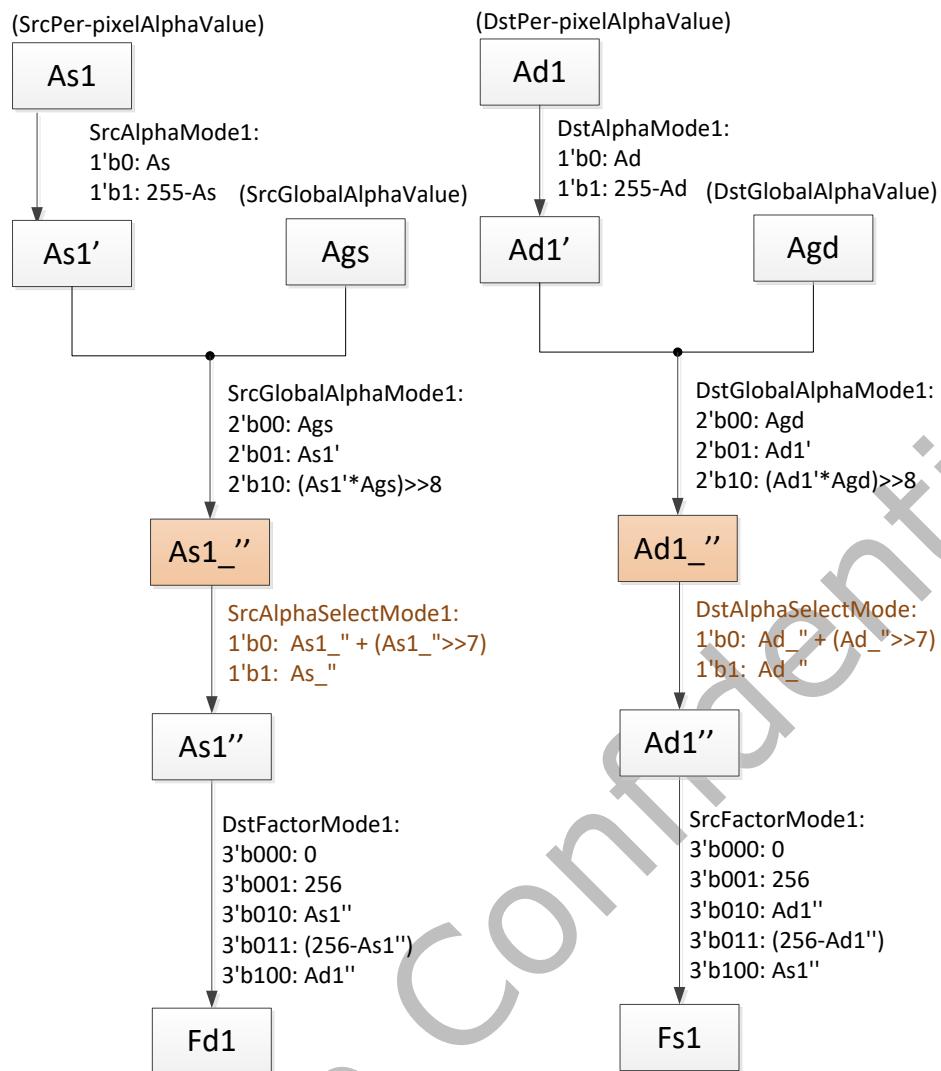


Fig. 7-5 Layer0 alpha blending calculate flow



$$Ad = Fs1 * As1'' + Fd1 * Ad1'' \quad (2)$$

(Ad – dst alpha, Fs1 – alpha src factor1, As1'' – src alpha'', Fd1 – alpha dst factor1, Ad1'' – dst alpha'')

Fig. 7-6 Layer1 alpha blending calculate flow

7.3.4 Color fill

Two modes of color fill can be done by RGA: solid fill and gradient fill.

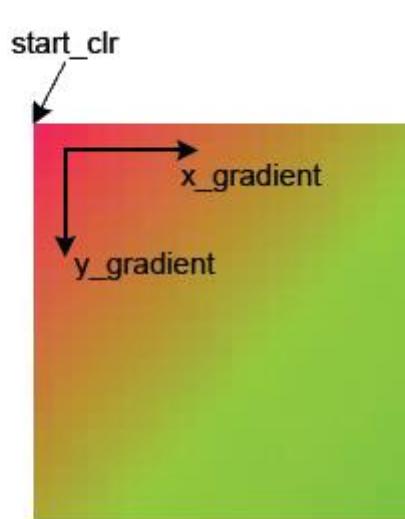


Fig. 7-7 RGA Gradient Fill

Gradient fill using following equations for ARGB calculation of every pixel in different

coordinary.

```
A_cur = (A_start + x*x_A_gradient) +y*y_A_gradient;
R_cur = (R_start + x*x_R_gradient) +y*y_R_gradient;
G_cur = (G_start + x*x_G_gradient) +y*y_G_gradient;
B_cur = (B_start + x*x_B_gradient) +y*y_B_gradient;
```

`A_start, R_start, G_start, B_start` is the ARGB value of start point. There are four pairs of values for horizontal and vertical gradient. Saturation operation could be enabled or disabled if the color overflows 255 or underflows 0.

7.3.5 Raster Operation (ROP)

Raster operation (ROP) is a Boolean operation between operands, which involve AND, OR, XOR, and NOT operations. For ROP2, operands are P (select pan) and D (Destination bitmap). For ROP3, operands are P (pattern), S (source bitmap) and D (Destination bitmap). For ROP4, operands are P (pattern), S (source bitmap), D (Destination bitmap) and MASK.

Table 7-1 RGAROP Boolean operations

| Operator | Meaning |
|----------|----------------------------|
| a | Bitwise AND |
| n | Bitwise NOT (inverse) |
| o | Bitwise OR |
| x | Bitwise exclusive OR (XOR) |

7.3.6 Scaling

The scaling operation is the imageresizing processingof source image. Scaling is done base on ARGB8888 format.

There are three scale modes: scale down (bilinear, Average); scale up(bilinear, Bi-cubic);

7.4 Register Description

7.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

7.4.2 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|--------------------------------------|--------|------|-------------|---|
| <u>RGA2_SYS_CTRL</u> | 0x0000 | W | 0x00000044 | RGA system control register |
| <u>RGA2_CMD_CTRL</u> | 0x0004 | W | 0x00000000 | RGA command control register |
| <u>RGA2_CMD_BASE</u> | 0x0008 | W | 0x12345678 | RGA command codes base address register |
| <u>RGA2_STATUS1</u> | 0x000c | W | 0x00000000 | RGA status register |
| <u>RGA2_INT</u> | 0x0010 | W | 0x00000000 | RGA interrupt register |
| <u>RGA2_MMU_CTRL0</u> | 0x0014 | W | 0x00000000 | Register0000 Description |
| <u>RGA2_MMU_CMD_BASE</u> | 0x0018 | W | 0x00000000 | Register0000 Description |
| <u>RGA2_STATUS2</u> | 0x001c | W | 0x00000000 | RGA status register |
| <u>RGA2_WORK_CNT</u> | 0x0020 | W | 0x00000000 | |
| <u>RGA2_VERSION_INFO</u> | 0x0028 | W | 0x00000000 | RTL version and FPGA version information |
| <u>RGA2_PERF_LATENCY_CT_RL0</u> | 0x0040 | W | 0x00000024 | Only exist when this IP has axi_performance monitor feature |
| <u>RGA2_PERF_LATENCY_CT_RL1</u> | 0x0044 | W | 0x00000021 | Only exist when this IP has axi_performance monitor feature |
| <u>RGA2_PERF_RD_MAX_LATENCY_NUM0</u> | 0x0048 | W | 0x00000000 | Only exist when this IP has axi_performance monitor feature |

| Name | Offset | Size | Reset Value | Description |
|-------------------------------|--------|------|-------------|---|
| RGA2 PERF RD LATENCY_SAMP_NUM | 0x004c | W | 0x00000000 | Only exist when this IP has axi_performance monitor feature |
| RGA2 PERF RD LATENCY_ACC_SUM | 0x0050 | W | 0x00000000 | Only exist when this IP has axi_performance monitor feature |
| RGA2 PERF RD AXI TOT_AL_BYTE | 0x0054 | W | 0x00000000 | Only exist when this IP has axi_performance monitor feature |
| RGA2 PERF WR AXI TOT_AL_BYTE | 0x0058 | W | 0x00000000 | Only exist when this IP has axi_performance monitor feature |
| RGA2 PERF WORKING_CNT | 0x005c | W | 0x00000000 | Only exist when this IP has axi_performance monitor feature |
| RGA2 MODE CTRL | 0x0100 | W | 0x00000000 | RGA mode control register |
| RGA2 SRC INFO | 0x0104 | W | 0x00000000 | RGA source information register |
| RGA2 SRC BASE0 | 0x0108 | W | 0x00000000 | RGA source image Y/RGB base address register |
| RGA2 SRC BASE1 | 0x010c | W | 0x00000000 | RGA source image Cb/Cbr base address register |
| RGA2 SRC BASE2 | 0x0110 | W | 0x00000000 | RGA source image Cr base address register |
| RGA2 SRC BASE3 | 0x0114 | W | 0x00000000 | RGA source image 1 base address register |
| RGA2 SRC VIR INFO | 0x0118 | W | 0x00000000 | RGA source image virtual stride / RGA source image tile number register |
| RGA2 SRC ACT INFO | 0x011c | W | 0x00000000 | RGA source image active width/height register |
| RGA2 SRC X FACTOR | 0x0120 | W | 0x00000000 | RGA source image horizontal scaling factor |
| RGA2 SRC Y FACTOR | 0x0124 | W | 0x00000000 | RGA source image vertical scaling factor |
| RGA2 SRC BG COLOR | 0x0128 | W | 0x00000000 | RGA source image background color |
| RGA2 SRC FG COLOR | 0x012c | W | 0x00000000 | RGA source image foreground color |
| RGA2 SRC TR COLOR0 | 0x0130 | W | 0x00000000 | RGA source image transparency color min value |
| RGA2 CP GR A | 0x0130 | W | 0x00000000 | RGA color gradient fill step register (color fill mode) |
| RGA2 SRC TR COLOR1 | 0x0134 | W | 0x00000000 | RGA source image transparency color max value |
| RGA2 CP GR B | 0x0134 | W | 0x00000000 | RGA color gradient fill step register (color fill mode) |
| RGA2 DST INFO | 0x0138 | W | 0x00000000 | RGA destination format register |

| Name | Offset | Size | Reset Value | Description |
|--------------------|--------|------|-------------|---|
| RGA2_DST_BASE0 | 0x013c | W | 0x00000000 | RGA destination image base address 0 register |
| RGA2_DST_BASE1 | 0x0140 | W | 0x00000000 | RGA destination image base address 1 register |
| RGA2_DST_BASE2 | 0x0144 | W | 0x00000000 | RGA destination image base address 2 register |
| RGA2_DST_VIR_INFO | 0x0148 | W | 0x00000000 | RGA destination image virtual width/height register |
| RGA2_DST_ACT_INFO | 0x014c | W | 0x00000000 | RGA destination image active width/height register |
| RGA2_ALPHA_CTRL0 | 0x0150 | W | 0x00000000 | Alpha control register 0 |
| RGA2_ALPHA_CTRL1 | 0x0154 | W | 0x00000000 | Register0000 Description |
| RGA2_FADING_CTRL | 0x0158 | W | 0x00000000 | Fading control register |
| RGA2_PAT_CON | 0x015c | W | 0x00000000 | Pattern size/offset register |
| RGA2_ROP_CON0 | 0x0160 | W | 0x00000000 | ROP code 0 control register |
| RGA2_CP_GR_G | 0x0160 | W | 0x00000000 | RGA color gradient fill step register (color fill mode) |
| RGA2_ROP_CON1 | 0x0164 | W | 0x00000000 | ROP code 1 control register |
| RGA2_CP_GR_R | 0x0164 | W | 0x00000000 | RGA color gradient fill step register (color fill mode) |
| RGA2_MASK_BASE | 0x0168 | W | 0x00000000 | RGA mask base address register |
| RGA2_MMU_CTRL1 | 0x016c | W | 0x00000000 | RGA MMU control register 1 |
| RGA2_MMU_SRC_BASE | 0x0170 | W | 0x00000000 | RGA source MMU TLB base address |
| RGA2_MMU_SRC1_BASE | 0x0174 | W | 0x00000000 | RGA source1 MMU TLB base address |
| RGA2_MMU_DST_BASE | 0x0178 | W | 0x00000000 | RGA destination MMU TLB base address |
| RGA2_MMU_ELS_BASE | 0x017c | W | 0x00000000 | RGA ELSE MMU TLB base address |

Notes:*B*- Byte (8 bits) access, *HW*- Half WORD (16 bits) access, *W*-WORD (32 bits) access

7.4.3 Detail Register Description

RGA2_SYS_CTRL

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | sw_RST_HANDSAVE_P it would save protect-rstn into initial status if long time dead in protect-rstn status. (auto clear into '0') |
| 6 | RW | 0x1 | sw_RST_PROTECT_E protect-rstn mode enable. it would be ensure all axi write/read operation into completion status when sw_CCLK_SRESET_P or sw_ACLK_SRESET_P valid |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 5 | RW | 0x0 | sw_auto_RST it would auto-reset after one frame finish. 0: disable 1: enable |
| 4 | RW | 0x0 | sw_cclk_sreset_p RGA core clk domain Soft reset, write '1' to this would reset the RGA engine except config registers |
| 3 | WO | 0x0 | sw_aclk_sreset_p RGA aclk domain Soft reset, write '1' to this would reset the RGA engine except config registers |
| 2 | WO | 0x1 | sw_auto_ckg RGA auto clock gating enable bit 0: disable 1: enable |
| 1 | WO | 0x0 | sw_cmd_mode RGA command mode 0: slave mode 1: master mode |
| 0 | W1C | 0x0 | sw_cmd_op_st_p Only used in passive (slave) control mode |

RGA2_CMD_CTRL

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | reserved |
| 12:3 | RW | 0x000 | sw_cmd_incr_num RGA command increment number |
| 2 | WO | 0x0 | sw_cmd_stop RGA command stop mode Command execution would stop after the current graphic operation finish if set this bit to 1 |
| 1 | WO | 0x0 | sw_cmd_incr_valid_p RGA command increment valid (Auto cleared) When setting this bit, 1. The total cmd number would increase by the RGA_INCR_CMD_NUM. 2. RGA would continue running if idle |
| 0 | RW | 0x0 | sw_cmd_line_st_p RGA command line fetch start (command line reset) (Auto cleared) When fetch start, the total cmd number would reset to RGA_INCR_CMD_NUM |

RGA2_CMD_BASE

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x12345678 | sw_cmd_base RGA command codes base address |

RGA2_STATUS1

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x000 | sw_cmd_total_num RGA command total number |
| 19:8 | RO | 0x000 | sw_cmd_cur_num RGA command current number |
| 7:1 | RO | 0x00 | Reserved Reserved |
| 0 | RO | 0x0 | sw_rga_sta RGA engine status 0: idle 1: working |

RGA2_INT

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | sw_intr_af_e All command finished interrupt enable |
| 9 | RW | 0x0 | sw_intr_mmu_e MMU interrupt enable |
| 8 | RW | 0x0 | sw_intr_err_e Error interrupt enable |
| 7 | WO | 0x0 | sw_intr_cf_clr Current command finished interrupt clear |
| 6 | WO | 0x0 | sw_intr_af_clr All command finished interrupt clear |
| 5 | WO | 0x0 | sw_intr_mmu_clr MMU interrupt clear |
| 4 | WO | 0x0 | sw_intr_err_clr Error interrupt clear |
| 3 | RO | 0x0 | sw_intr_cf Current command finished interrupt flag |
| 2 | RO | 0x0 | sw_intr_af All command finished interrupt flag |
| 1 | RO | 0x0 | sw_intr_mmu MMU interrupt |
| 0 | RO | 0x0 | sw_intr_err Error interrupt flag |

RGA2_MMU_CTRL0

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RW | 0x0000000 | Reserved Reserved |
| 10:9 | RW | 0x0 | sw_els_ch_priority sw_els_ch_priority |
| 8:7 | RW | 0x0 | sw_dst_ch_priority sw_dst_ch_priority |
| 6:5 | RW | 0x0 | sw_src1_ch_priority sw_src1_ch_priority |
| 4:3 | RW | 0x0 | sw_src_ch_priority sw_src_ch_priority |
| 2 | RW | 0x0 | sw_cmd_mmu_flush RGA CMD channel MMU TLB flush: Set 1 to this bit to flush MMU TLB, auto clear |
| 1 | RW | 0x0 | sw_cmd_mmu_en RGA CMD channel MMU enable 0: disable 1: enable |
| 0 | RW | 0x0 | sw_mmu_page_size RGA MMU Page table size 0: 4KB page 1: 64KB page |

RGA2_MMU_CMD_BASE

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:0 | RW | 0x0000000 | sw_mmu_cmd_base RGA command MMU TLB base address (word) |

RGA2_STATUS2

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:13 | RO | 0x00000 | Reserved |
| 12:11 | RO | 0x0 | rpp_mkram_rready rpp_mkram_rready |
| 10:6 | RO | 0x00 | dstrpp_outbuf_rready dstrpp_outbuf_rready |
| 5:2 | RO | 0x0 | srcrpp_outbuf_rready dstrpp_outbuf_rready |
| 1 | RO | 0x0 | bus_error |
| 0 | RO | 0x0 | rpp_error |

RGA2 WORK CNT

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RW | 0x00 | Reserved |
| 26:0 | RO | 0x00000000 | sw_work_cnt RGA total working counter |

RGA2 VERSION INFO

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0x00 | major used for IP structure version infomation |
| 23:20 | RW | 0x0 | minor big feature change under same structure |
| 19:0 | RW | 0x00000 | svnbuid rtl current svn number |

RGA2 PERF LATENCY CTRL0

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:8 | RW | 0x000 | sw_rd_latency_thr |
| 7:4 | RW | 0x2 | sw_rd_latency_id |
| 3 | RW | 0x0 | sw_axi_cnt_type sw_axi_cnt_type |
| 2 | RW | 0x1 | sw_axi_perf_frm_type 1'b0: clear by software configuration 1'b1: clear by frame end |
| 1 | RW | 0x0 | sw_axi_perf_clr_e 1'b0: software clear disable 1'b1: software clear enable |
| 0 | RW | 0x0 | sw_axi_perf_work_e 1'b0: disable 1'b1: enable |

RGA2 PERF LATENCY CTRL1

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x0 | sw_aw_count_id sw_aw_count_id |
| 7:4 | RW | 0x2 | sw_ar_count_id sw_ar_count_id |
| 3 | RW | 0x0 | sw_aw_cnt_id_type sw_aw_cnt_id_type |
| 2 | RW | 0x0 | sw_ar_cnt_id_type sw_ar_cnt_id_type |
| 1:0 | RW | 0x1 | sw_addr_align_type sw_addr_align_type |

RGA2 PERF RD MAX LATENCY NUM0

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:16 | RO | 0x0 | reserved |
| 15:0 | RO | 0x0000 | rd_max_latency_num_ch0 |

RGA2 PERF RD LATENCY SAMP NUM

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:0 | RO | 0x00000000 | rd_latency_thr_num_ch0 |

RGA2 PERF RD LATENCY ACC SUM

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:0 | RO | 0x00000000 | rd_latency_acc_sum |

RGA2 PERF RD AXI TOTAL BYTE

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | perf_rd_axi_total_byte perf_rd_axi_total_byte |

RGA2 PERF WR AXI TOTAL BYTE

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | perf_wr_axi_total_byte perf_wr_axi_total_byte |

RGA2 PERF WORKING CNT

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------------|
| 31:0 | RW | 0x00000000 | perf_working_cnt perf_working_cnt |

RGA2 MODE CTRL

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RW | 0x0000000 | Reserved Reserved |
| 7 | RW | 0x0 | sw_intr_cf_e Current command finished interrupt enable |
| 6 | RW | 0x0 | sw_gradient_sat Gradient saturation calculation mode 0:clip 1:not-clip |
| 5 | RW | 0x0 | sw_alpha_zero_key ARGB888 alpha zero key mode 0x000000 would be changed to 0x000100(RGB888)/0x0020(RGB565)for ARGB888 to RGBX/RGB565 color key 0: disable 1: enable |
| 4 | RW | 0x0 | sw_cf_rop4_pat Color fill/ROP4 pattern 0: solid color 1: pattern color |
| 3 | RW | 0x0 | sw_bb_mode Bitblt mode 0: SRC + DST => DST 1: SRC + SRC1 => DST |
| 2:0 | RW | 0x0 | sw_render_mode RGA 2D render mode 000: Bitblt 001: Color palette 010: Rectangle fill 011: Update palette LUT/pattern ram |

RGA2_SRC_INFO

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:29 | RO | 0x0 | reserved |
| 28 | RW | 0x0 | sw_src_yuv10_round_e this bit valid when RGA support yuv 10bit picture input 0: yuv 10bit to 8bit round disable 1: yuv 10bit to 8bit round enable |
| 27 | RW | 0x0 | sw_src_yuv10_e this bit valid when RGA support yuv 10bit picture input 0: yuv 10bit disable 1: yuv 10bit enable |
| 26 | RW | 0x0 | sw_vsp_mode 0:by-cubic 1:bi-linear |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 25:24 | RW | 0x0 | sw_bic_coe_sel SRC bicubic scaling coefficient select 00: CATROM 01: MITCHELL 10: HERMITE 11: B-SPLINE |
| 23 | RW | 0x0 | sw_src_dither_up SRC dither up enable 0:disable 1:enable |
| 22:19 | RW | 0x0 | sw_src_trans_e Source transparency enable bits [3]: A value stencil test enable bit [2]: B value stencil test enable bit [1]: G value stencil test enable bit [0]: R value stencil test enable bit |
| 18 | RW | 0x0 | sw_src_trans_mode Source transparency mode 0: normal stencil test (color key) 1: inverted stencil test |
| 17:16 | RW | 0x0 | sw_src_vscl_mode SRC vertical scaling mode 00: no scaling 01: down-scaling 10: up-scaling |
| 15:14 | RW | 0x0 | sw_src_hscl_mode SRC horizontal scaling mode 00: no scaling 01: down-scaling 10: up-scaling |
| 13:12 | RW | 0x0 | sw_src_mir_mode SRC mirror mode 00: no mirror 01: x mirror 10: y mirror 11: x mirror + y mirror |
| 11:10 | RW | 0x0 | sw_src_rot_mode SRC rotation mode 00: 0 degree 01: 90 degree 10: 180 degree 11: 270 degree |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9:8 | RW | 0x0 | sw_src_csc_mode Source bitmap YUV2RGB conversion mode 00: bypass 01: BT.601-range0 10: BT.601-range1 11: BT.709-range0 |
| 7 | RW | 0x0 | sw_cp_endian Source Color palette endian swap 0: big endian 1: little endian |
| 6 | RW | 0x0 | sw_src_uvswap Source Cb-Cr swap 0: CrCb 1: CbCr |
| 5 | RW | 0x0 | sw_src_alpha_swap Source bitmap data alpha swap 0: ABGR 1: BGRA |
| 4 | RW | 0x0 | sw_src_rbswap Source bitmap data RB swap 0: BGR 1: RGB |
| 3:0 | RW | 0x0 | sw_src_fmt Source bitmap data format 0000: ABGR888 0001: XBGR888 0010: BGR packed 0100: RGB565 0101: ARGB1555 0110: ARGB4444 1000: YUV422SP 1001: YUV422P 1010: YUV420SP 1011: YUV420P 1100: 1BPP (color palette) 1101: 2BPP (color palette) 1110: 4BPP (color palette) 1111: 8BPP (color palette) |

RGA2_SRC_BASE0

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_src_base0 source image Y/RGB base address |

RGA2_SRC_BASE1

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_src_base1 source image Cb base address (YUV422/420-P) source image Cb/Cr base address (YU,V422/420-SP) |

RGA2_SRC_BASE2

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | sw_src_base2 source image Cr base address (YUV422/420-P) |

RGA2_SRC_BASE3

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | sw_src_base3 source image 1 RGB base address (source bitblt mode1) |

RGA2_SRC_VIR_INFO

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:26 | RW | 0x00 | Reserved |
| 25:16 | RW | 0x000 | sw_mask_vir_stride mask image virtual stride (words) |
| 15 | RW | 0x0 | Reserved Reserved |
| 14:0 | RW | 0x0000 | sw_src_vir_stride src image virtual stride (words) |

RGA2_SRC_ACT_INFO

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RW | 0x0 | Reserved2 Reserved |
| 28:16 | RW | 0x0000 | sw_src_act_height source image active height |
| 15:13 | RW | 0x0 | Reserved1 Reserved |
| 12:0 | RW | 0x0000 | sw_src_act_width source image active width |

RGA2_SRC_X_FACTOR

Address: Operational Base + offset (0x0120)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | sw_src_hsp_factor Source image horizontal up-scaling factor =(DST_ACT_WIDTH/SRC_ACT_WIDTH) * 65536 |
| 15:0 | RW | 0x0000 | sw_src_hsd_factor Source image horizontal down-scaling factor =(SRC_ACT_WIDTH/DST_ACT_WIDTH) * 65536 |

RGA2_SRC_Y_FACTOR

Address: Operational Base + offset (0x0124)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | sw_src_vsp_factor Source image vertical up-scaling factor (DST_ACT_HEIGHT/SRC_ACT_HEIGHT) * 65536 |
| 15:0 | RW | 0x0000 | sw_src_vsd_factor Source image vertical down-scaling factor (SRC_ACT_HEIGHT/DST_ACT_HEIGHT) * 65536 |

RGA2_SRC_BG_COLOR

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_src_bg_color Source image background color ("0" bit color for mono expansion.) |

RGA2_SRC_FG_COLOR

Address: Operational Base + offset (0x012c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | sw_src_fg_color Source image foreground color ("1" bit color for mono expansion.) Color fill color, Pan color |

RGA2_SRC_TR_COLOR0

Address: Operational Base + offset (0x0130)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x00 | sw_src_trans_amin source image transparency color A min value |
| 23:16 | RW | 0x00 | sw_src_trans_bmin source image transparency color B min value |
| 15:8 | RW | 0x00 | sw_src_trans_gmin source image transparency color G min value |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7:0 | RW | 0x00 | sw_src_trans_rmin source image transparency color R min value |

RGA2 CP GR A

Address: Operational Base + offset (0x0130)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | sw_gradient_y_a Y gradient value of Alpha (signed 8.8) |
| 15:0 | RW | 0x0000 | sw_gradient_x_a X gradient value of Alpha (signed 8.8) |

RGA2 SRC TR COLOR1

Address: Operational Base + offset (0x0134)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x00 | sw_src_trans_amax source image transparency color A max value |
| 23:16 | RW | 0x00 | sw_src_trans_bmax source image transparency color B max value |
| 15:8 | RW | 0x00 | sw_src_trans_gmax source image transparency color G max value |
| 7:0 | RW | 0x00 | sw_src_trans_rmax source image transparency color R max value |

RGA2 CP GR B

Address: Operational Base + offset (0x0134)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | sw_gradient_y_b Y gradient value of Blue (signed 8.8) |
| 15:0 | RW | 0x0000 | sw_gradient_x_b X gradient value of Blue (signed 8.8) |

RGA2 DST INFO

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21 | RW | 0x0 | sw_src1_csc_clip src1 read BGR2YUV Clip mode(from 0~255 clip to 36~235) 1: clip enable; 0: unclip |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 20:19 | RW | 0x0 | sw_src1_csc_mode sw_dst_csc_mode SRC1 read bitmap RGB2YUV conversion mode 00: Bypass 01: BT.601-range0 10: BT.601-range1 11: BT.709-range0 |
| 18 | RW | 0x0 | sw_dst_csc_clip dst write BGR2YUV Clip mode(from 0~255 clip to 36~235) 1: clip enable; 0: unclip |
| 17:16 | RW | 0x0 | sw_dst_csc_mode sw_dst_csc_mode DST write bitmap RGB2YUV conversion mode 00: Bypass 01: BT.601-range0 10: BT.601-range1 11: BT.709-range0 |
| 15:14 | RW | 0x0 | sw_dither_mode sw_dither_mode DST dither down bit mode 00: 888 to 666 01: 888 to 565 10: 888 to 555 11: 888 to 444 |
| 13 | RW | 0x0 | sw_dither_down sw_dither_down DST dither down enable 0:disable 1:enable |
| 12 | RW | 0x0 | sw_src1_dither_up sw_src1_dither_up DST/SRC1 dither up enable 0:disable 1:enable |
| 11 | RW | 0x0 | sw_src1_alpha_swap sw_src1_alpha_swap Source 1 bitmap data alpha swap 0: ABGR 1: BGRA |
| 10 | RW | 0x0 | sw_src1_rbswap sw_src1_rbswap Source 1 bitmap data RB swap 0: BGR 1: RGB |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9:7 | RW | 0x0 | <p>sw_src1_fmt Source 1 bitmap data format 000: ABGR888 001: XBGR888 010: BGR packed 100: RGB565 101: ARGB1555 110: ARGB4444</p> |
| 6 | RW | 0x0 | <p>sw_dst_uvswap Destination Cb-Cr swap 0: CrCb 1: CbCr</p> |
| 5 | RW | 0x0 | <p>sw_dst_alpha_swap Destination bitmap data alpha swap 0: ABGR 1: BGRA</p> |
| 4 | RW | 0x0 | <p>sw_dst_rbswap Destination bitmap data RB swap 0: BGR 1: RGB</p> |
| 3:0 | RW | 0x0 | <p>sw_dst_fmt Destination bitmap data format 0000: ABGR888 0001: XBGR888 0010: BGR packed 0100: RGB565 0101: ARGB1555 0110: ARGB4444 1000: YUV422SP 1001: YUV422P 1010: YUV420SP 1011: YUV420P If RGA has yuyv output format feature: 1100: YVYU422(U, LSB) 1101: YVYU420(U, LSB) 1110: VYUY422(Y, LSB) 1111: VYUY420(Y, LSB)</p> |

RGA2 DST BASE0

Address: Operational Base + offset (0x013c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | sw_dst_base0 destination image Y/RGB base address |

RGA2 DST BASE1

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | sw_dst_base1 destination image Cb/CbCr base address |

RGA2 DST BASE2

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | sw_dst_base2 destination image Cr base address |

RGA2 DST VIR INFO

Address: Operational Base + offset (0x0148)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RW | 0x0 | Reserved2 Reserved |
| 27:16 | RW | 0x000 | sw_src1_vir_stride source image 1 virtual stride (words) |
| 15:12 | RW | 0x0 | Reserved1 Reserved |
| 11:0 | RW | 0x000 | sw_dst_vir_stride destination image virtual stride(words) |

RGA2 DST ACT INFO

Address: Operational Base + offset (0x014c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RW | 0x0 | Reserved2 Reserved |
| 27:16 | RW | 0x000 | sw_dst_act_height Destination image active height |
| 15:12 | RW | 0x0 | Reserved1 Reserved |
| 11:0 | RW | 0x000 | sw_dst_act_width Destination image active width |

RGA2 ALPHA CTRL0

Address: Operational Base + offset (0x0150)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------|
| 31:21 | RW | 0x000 | Reserved Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 20 | RW | 0x0 | sw_mask_endian ROP4 mask endian swap 0: big endian 1: little endian |
| 19:12 | RW | 0x00 | sw_dst_global_alpha global alpha value of DST(Agd) |
| 11:4 | RW | 0x00 | sw_src_global_alpha global alpha value of SRC(Ags) fading value in fading mod |
| 3:2 | RW | 0x0 | sw_rop_mode ROP mode select 00: ROP 2 01: ROP 3 10: ROP 4 |
| 1 | RW | 0x0 | sw_alpha_rop_sel Alpha or ROP select 0: alpha 1: ROP |
| 0 | RW | 0x0 | sw_alpha_rop_e Alpha or ROP enable 0: disable 1: enable |

RGA2 ALPHA CTRL1

Address: Operational Base + offset (0x0154)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RW | 0x0 | Reserved Reserved |
| 29 | RW | 0x0 | sw_src_alpha_m1 Src Transparent/opaque of alpha channel (As1') 0: As 1: 255-As |
| 28 | RW | 0x0 | sw_dst_alpha_m1 Dst Transparent/opaque of alpha channel (Ad1') 0: Ad 1: 255-Ad |
| 27:26 | RW | 0x0 | sw_src_blend_m1 Alpha src blend mode select of alpha channel (As1_) 00: Ags 01: As1' 10: (As1'*Ags)>>8 11: reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 25:24 | RW | 0x0 | sw_dst_blend_m1 Alpha dst blend mode select of alpha channel(Ad1_") 00: Agd 01: Ad1' 10: (Ad1'*Agd)>>8 11: reserved |
| 23 | RW | 0x0 | sw_src_alpha_cal_m1 Alpha src calculate mode of alpha channel(As1") 0: As1"= As1_"+ (As1_">>>7) 1: As1"= As1 _" |
| 22 | RW | 0x0 | sw_dst_alpha_cal_m1 Alpha dst calculate mode of alpha channel(Ad1") 0: Ad1"= Ad1_ " + (Ad1_">>>7) 1: Ad1"= Ad1_ " |
| 21:19 | RW | 0x0 | w_src_factor_m1 Src factore mode of alpha channel(Fs1) 000: 0 001: 256 010: Ad1" 011: 256-Ad1" 100: As1" |
| 18:16 | RW | 0x0 | sw_dst_factor_m1 Dst factore mode of alpha channel(Fd1) 000: 0 001: 256 010: As1" 011: 256-As1" 100: Ad1" |
| 15 | RW | 0x0 | sw_src_alpha_m0 Src Transparent/opaque of color channel (As0') 0: As 1: 255-As |
| 14 | RW | 0x0 | sw_dst_alpha_m0 Dst Transparent/opaque of color channel (Ad0') 0: Ad 1: 255-Ad |
| 13:12 | RW | 0x0 | sw_src_blend_m0 Alpha src blend mode select of color channel (As0_") 00: Ags 01: As0' 10: (As0'*Agss)>>8 11: reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11:10 | RW | 0x0 | sw_dst_blend_m0 Alpha dst blend mode select of color channel(Ad0_) 00: Agd 01: Ad0' 10: (Ad0'*Agd)>>8 11: reserved |
| 9 | RW | 0x0 | sw_src_alpha_cal_m0 Alpha src calculate mode of color channel(As0'') 0: As0''= As0_''+ (As0_''>>7) 1: As0''= As0_'' |
| 8 | RW | 0x0 | sw_dst_alpha_cal_m0 Alpha dst calculate mode of color channel(Ad0'') 0: Ad0''= Ad0_'' + (Ad0_''>>7) 1: Ad0''= Ad0_'' |
| 7:5 | RW | 0x0 | sw_src_factor_m0 Src factore mode of color channel(Fs0) 000: 0 001: 256 010: Ad0'' 011: 256-Ad0'' 100: As0'' |
| 4:2 | RW | 0x0 | sw_dst_factor_m0 Dst factore mode of color channel(Fd0) 000: 0 001: 256 010: As0'' 011: 256-As0'' 100: Ad0'' |
| 1 | RW | 0x0 | sw_src_color_m0 SRC color select(Cs') 0: Cs 1: Cs * As0'' |
| 0 | RW | 0x0 | sw_dst_color_m0 SRC color select(Cd') 0: Cd 1: Cd * Ad0'' |

RGA2 FADING CTRL

Address: Operational Base + offset (0x0158)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------|
| 31:25 | RW | 0x00 | Reserved Reserved |
| 24 | RW | 0x0 | sw_fading_en Fading enable |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23:16 | RW | 0x00 | sw_fading_offset_b Fading offset B value |
| 15:8 | RW | 0x00 | sw_fading_offset_g Fading offset G value (Pattern total number when pattern loading) |
| 7:0 | RW | 0x00 | sw_fading_offset_r Fading offset R value (Start point of pattern ram in pattern mode) |

RGA2 PAT CON

Address: Operational Base + offset (0x015c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:24 | RW | 0x00 | sw_pat_offset_y Pattern y offset |
| 23:16 | RW | 0x00 | sw_pat_offset_x Pattern x offset |
| 15:8 | RW | 0x00 | sw_pat_height Pattern height |
| 7:0 | RW | 0x00 | sw_pat_width Pattern width |

RGA2 ROP CON0

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RW | 0x00 | Reserved Reserved |
| 24:0 | RW | 0x00000000 | sw_rop3_code0 Rop3 code 0 control bits |

RGA2 CP GR G

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | sw_gradient_y_g Y gradient value of Green (signed 8.8) |
| 15:0 | RW | 0x0000 | sw_gradient_x_g X gradient value of Green (signed 8.8) |

RGA2 ROP CON1

Address: Operational Base + offset (0x0164)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RW | 0x00 | Reserved Reserved |
| 24:0 | RW | 0x00000000 | sw_rop3_code1 Rop3 code 1 control bits |

RGA2 CP GR R

Address: Operational Base + offset (0x0164)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | sw_gradient_y_r Y gradient value of Red(signed 8.8) |
| 15:0 | RW | 0x0000 | sw_gradient_x_r X gradient value of Red(signed 8.8) |

RGA2 MASK BASE

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | sw_mask_base mask base address in ROP4 mode LUT/ pattern load base address |

RGA2 MMU CTRL1

Address: Operational Base + offset (0x016c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:14 | RW | 0x00000 | Reserved Reserved |
| 13 | RW | 0x0 | sw_els_mmu_flush RGA ELSE channel MMU TLB flush: Set 1 to this bit to flush MMU TLB, auto clear |
| 12 | RW | 0x0 | sw_els_mmu_en RGA ELSE channel MMU enable 0: disable 1: enable |
| 11 | RW | 0x0 | sw_dst_mmu_prefetch_dir sw_dst_mmu_prefetch_dir 0:forward 1:backward |
| 10 | RW | 0x0 | sw_dst_mmu_prefetch_en sw_dst_mmu_prefetch_en 0:disable 1:enable |
| 9 | RW | 0x0 | sw_dst_mmu_flush sw_dst_mmu_flush RGA DST channel MMU TLB flush: Set 1 to this bit to flush MMU TLB, auto clear |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 8 | RW | 0x0 | sw_dst_mmu_en sw_dst_mmu_en RGA DST channel MMU enable 0: disable 1: enable |
| 7 | RW | 0x0 | sw_src1_mmu_prefetch_dir sw_src1_mmu_prefetch_dir 0:forward 1:backward |
| 6 | RW | 0x0 | sw_src1_mmu_prefetch_en sw_src1_mmu_prefetch_en 0:disable 1:enable |
| 5 | RW | 0x0 | sw_src1_mmu_flush sw_src1_mmu_flush RGA SRC1 channel MMU TLB flush: Set 1 to this bit to flush MMU TLB, auto clear |
| 4 | RW | 0x0 | sw_src1_mmu_en sw_src1_mmu_en RGA SRC1 channel MMU enable 0: disable 1: enable |
| 3 | RW | 0x0 | sw_src_mmu_prefetch_dir sw_src_mmu_prefetch_dir 0:forward 1:backward |
| 2 | RW | 0x0 | sw_src_mmu_prefetch_en sw_src_mmu_prefetch_en 0:disable 1:enable |
| 1 | RW | 0x0 | sw_src_mmu_flush RGA SRC channel MMU TLB flush: Set 1 to this bit to flush MMU TLB, auto clear |
| 0 | RW | 0x0 | sw_src_mmu_en RGA SRC channel MMU enable 0: disable 1: enable |

RGA2 MMU SRC BASE

Address: Operational Base + offset (0x0170)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:0 | RW | 0x00000000 | sw_mmu_src_base RGA source MMU TLB base address (128-bit) |

RGA2_MMU_SRC1_BASE

Address: Operational Base + offset (0x0174)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:0 | RW | 0x00000000 | sw_mmu_src1_base RGA source1 MMU TLB base address (128-bit) |

RGA2_MMU_DST_BASE

Address: Operational Base + offset (0x0178)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:0 | RW | 0x00000000 | sw_mmu_dst_base RGA destination MMU TLB base address (128-bit) |

RGA2_MMU_ELS_BASE

Address: Operational Base + offset (0x017C)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:0 | RW | 0x00000000 | sw_mmu_els_base RGA destination MMU TLB base address (128-bit) |

7.5 Application Notes

7.5.1 Register Partition

There are two types of register in RGA. The first 8 registers (0x0 - 0x1C) are general registers for system configuration including command mode, command parameter, RGA status, general interrupts. The other registers (from 0x100) are command registers for command codes.

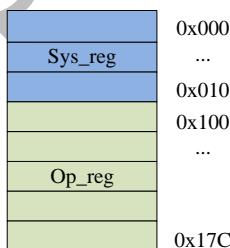


Fig. 7-8 RGAsoftware main register-region

7.5.2 Command Modes

RGA has two command modes: slave mode and master mode. In slave mode (RGA_SYS_CTRL[1] = 1'b0), 2D graphic command only could be run one by one. CPU set all the command registers in RGA and then start RGA running by setting RGA_SYS_CTRL[0] to '1'. In master mode (RGA_SYS_CTRL[1] = 1'b1), 2D graphic commands could be run sequentially. After setting command's number to RGA_CMD_CTRL[12:3], writing '1' to RGA_CMD_CTRL[0] will start the command fetch, then Internal command DMA fetch commands from external command line.

Command line is a collection of several command codes with continuous address. At the first start, the command start address (RGA_CMD_ADDR) and command number (RGA_CMD_CTRL[12:3]) should be set, then write '1' to cmd_line_st (RGA_CMD_CTRL[0]) to start the command line fetch. Incremental command is supported by setting cmd_incr_num (RGA_CMD_CTRL[12:3]) and cmd_incr_valid (RGA_CMD_CTRL[1]=1'b1)

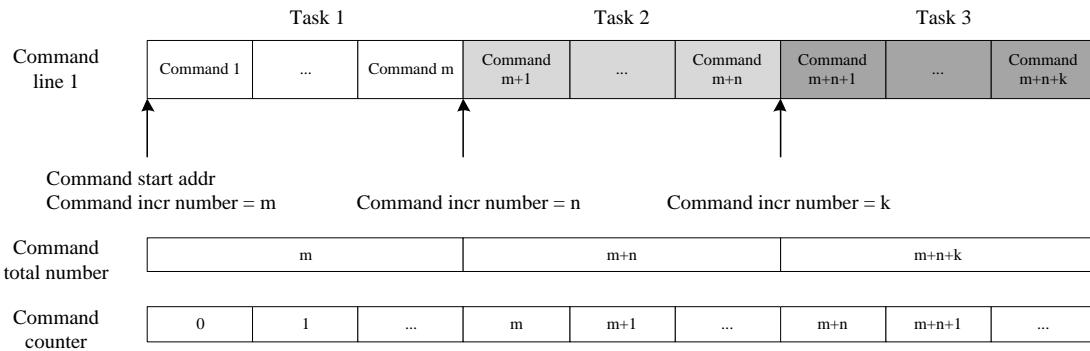


Fig. 7-9 RGA command line and command counter

7.5.3 Command Sync

In slave command mode, command sync is controlled by CPU.

In master command mode, user can enable the current_cmd_int(sw_intr_cf), command by command to generate a interrupt at the end point of target command operation.

| | | | | | | | | | |
|-----------|----------|----------------------------|----------|-----------------------------|----------|----------------------------|----------|-----------|----------|
| Command 1 | Run time | Command 2 (Intr enable) | Run time | Command 3 (Intr disable) | Run time | Command 4 (Intr enable) | Run time | Command 5 | Run time |
|-----------|----------|----------------------------|----------|-----------------------------|----------|----------------------------|----------|-----------|----------|

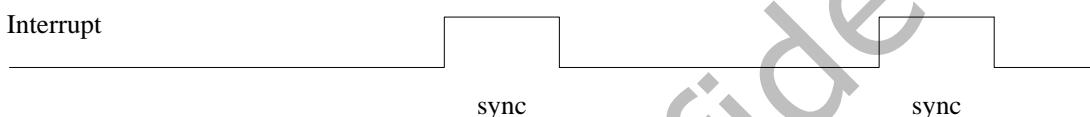


Fig. 7-10 RGA command sync generation

7.5.4 ColorPaletteApplication Notes

1. Palette/LUT Load into special RAM in ELS_BUF_CTRL;

2. ColorPalette/Pattern interval operations no need to initialLUT/pattern ram if LUT/patterncontentno update;

7.5.5 Some special application constraint

1. The algorithm of vertical scale up: must select bicubic algorithm when source picture is smaller or equal to 2k and must select bilinear when bigger than 2k
2. The effects that the output's definition is near 2k or 4k maybe not very well when at the scenario that the vertical side is scale up and the horizontal is scale down within range of 2%(such as:2048x32→2008x64)
3. At the scenario A+B->C, the size among the A B C has some constraint : A's size must be equal to C. C's size must equal to B when A+C is no rotation. C's rotation (90degree)size must equal to B1 when A+C is rotation 90 degree .

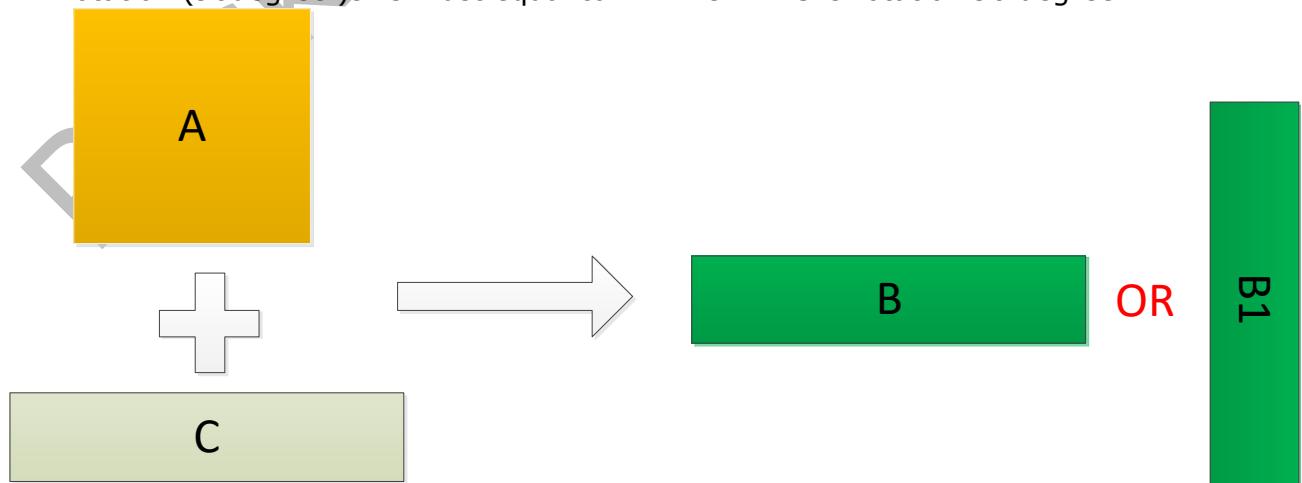


Fig. 7-11 The size constraint among A B C

4. YUV420/422-8bit virtual stride need 8byte align, xoff/yoff need 2byte algin;
5. YUV420/422-10bit virtual stride need 16byte align, not support xoff/yoff;
6. Vertical scale down or not && Horizontal Bi-cubic scale up src0 width<=2048;
Vertical scale up && Horizontal Bi-cubic scale up src0 width<=1928;
7. Vertical scale down or not && Horizontal Bilinear scale up src0 width<=4096;
Vertical scale up && Horizontal Bilinear scale up src0 width<=3856;

Rockchip Confidential

Chapter 8 USB2.0 Host Controller

8.1 Overview

USB2.0 host controller supports fully USB2.0 functions with one EHCI host controller and one OHCI host controller, and each host controller has one USB port. OHCI host controller only supports full-speed and low-speed mode and is used for full-speed devices and low-speed devices. EHCI only supports high-speed mode and is used for high-speed devices. OHCI host controller and EHCI host controller shares the same USB port, EHCI host controller will auto select the owner (OHCI or EHCI) of this USB port depending on the speed mode of attached devices, when selecting OHCI as owner, OHCI host controller will serve for the attached device; when selecting EHCI as owner, EHCI host controller will serve for the attached device.

USB2.0 Host Controller supports the following features:

- Compatible Specifications
 - Universal Serial Bus Specification, Revision 2.0
 - Enhanced Host Controller Interface Specification (EHCI), Revision 1.0
 - Open Host Controller Interface Specification (OHCI), Revision 1.0a
- Support High-speed (480Mbps), Full-speed (12Mbps) and Low-speed (1.5Mbps)

8.2 Block Diagram

USB2.0 Host Controller comprises with:

- EHCI Host Controller: Perform High-speed transactions
- OHCI Host Controller: Perform full/low-speed transactions
- Port Routing Control: Select EHCI Host Controller or OHCI Host Controller

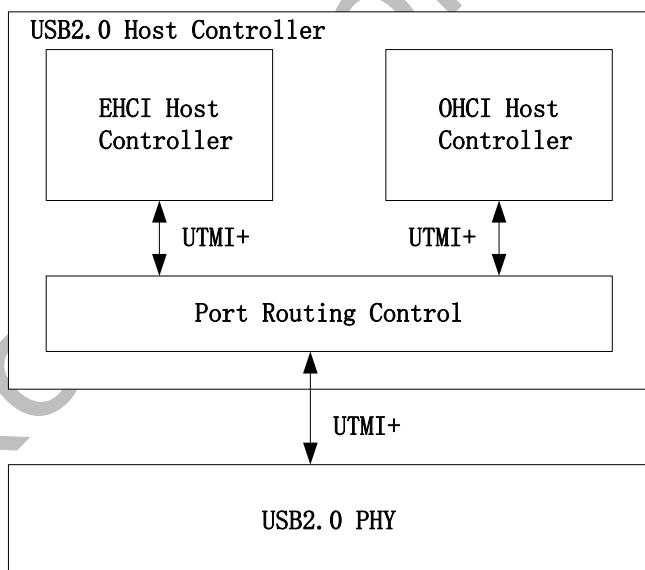


Fig. 8-1 USB2.0 Host Controller Block Diagram

8.3 Function Description

8.3.1 EHCI Host Controller

It performs descriptors and data read or write from or to system memory and packs or unpack USB transactions from or to UTMI+ interface defined in EHCI specification for high-speed data transmission.

8.3.2 OHCI Host Controller

It performs descriptors and data read/write from/to system memory and packs or un-pack USB transactions from or to UTMI+ interface defined in OHCI specification for full-speed or low-speed data transmission.

8.3.3 Port Routing Control

As part of logic in the EHCI host controller, it is used to auto-select EHCI or OHCI host controller to serve the attached device depending on the speed of the attached device.

8.4 Register Description

8.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

Table 8-1 USB2.0 Host Controller Address Mapping

| Base Address[16] | Device | Address Length | Offset Address Range |
|------------------|--------|----------------|----------------------|
| 1'b0 | EHCI | 64K BYTE | 0x00000 ~ 0xfffff |
| 1'b1 | OHCI | 64K BYTE | 0x10000 ~ 0x1fffff |

EHCI and OHCI register definitions, please refer to Enhanced Host Controller Interface Specification (EHCI), Revision 1.0 and Open Host Controller Interface Specification (OHCI), Revision 1.0a.

8.5 Interface Description

Table 8-2 USB2.0 PHY Interface Description

| Module Pin | Direction | Pin Name | Descriptions |
|------------|-----------|--------------|---|
| USB0ID | I | USB_OTG_ID | USB2.0 PHY OTG Port ID, left unused for TypeC |
| USB0PN | I/O | USB_OTG_DM | USB2.0 PHY OTG Port PN |
| USB0PP | I/O | USB_OTG_DP | USB2.0 PHY OTG Port PP |
| VBUS | I | USB_OTG_VBUS | USB2.0 PHY OTG Port VBUS |
| USB1PN | I/O | USB_HOST_DM | USB2.0 PHY Host Port PN |
| USB1PP | I/O | USB_HOST_DP | USB2.0 PHY Host Port PP |

8.6 Application Notes

8.6.1 Special Setting

8.6.2 Program flow

Please refer to Enhanced Host Controller Interface Specification (EHCI), Revision 1.0 and Open Host Controller Interface Specification (OHCI), Revision 1.0a.

Chapter 9 USB3.0 Controller

9.1 Overview

USB3.0 OTG Controller can act as static host, static device, USB2.0/3.0 OTG A device or B device basing on the status of input ID from USB2.0 PHY. It can perform data transmission between host and device as host or device for Super-Speed / High-Speed / Full-Speed / Low-Speed.

USB3.0 OTG controller supports the following features:

- General Features
 - 1. Compatible Specification
 - ◆ Universal Serial Bus 3.0 Specification, Revision 1.0
 - ◆ Universal Serial Bus Specification, Revision 2.0
 - ◆ eXtensible Host Controller Interface for Universal Serial Bus (xHCI), Revision 1.1
 - 2. Support Control/Bulk(including stream)/Interrupt/Isochronous Transfer
 - 3. Simultaneous IN and OUT transfer for USB3.0, up to 8Gbps bandwidth
 - 4. Descriptor caching and data pre-fetching used to improve system performance in high-latency systems
 - 5. LPM protocol in USB 2.0 and U0, U1, U2, and U3 states for USB 3.0
 - 6. Dynamic FIFO memory allocation for endpoints
 - 7. Keep-Alive feature in LS mode and (micro-)SOFs in HS/FS modes
 - 8. Low MIPS requirement
 - ◆ Driver involved only in setting up transfers and high-level error recovery
 - ◆ Hardware handles data packing and routing to a specific pipe
- Application Interface Features
 - 1. AHB Slave interface
 - 2. AXI Master interface
 - ◆ Programmable burst lengths up to 16
 - ◆ Handle fixed burst address alignment
 - ◆ Programmable number of outstanding read/write requests up to 16
 - ◆ Concurrent read/write to get best performance of USB3.0 duplex operation
- USB3.0 Device Features
 - 1. Up to 10 IN endpoints, including control endpoint 0
 - 2. Up to 6 OUT endpoints, including control endpoint 0
 - 3. Up to 16 endpoint transfer resources, each one for each endpoint
 - 4. Flexible endpoint configuration for multiple applications/USB set-configuration modes
 - 5. Hardware handles ERDY and burst
 - 6. Stream-based bulk endpoints with controller automatically initiating data movement
 - 7. Isochronous endpoints with isochronous data in data buffers
 - 8. Flexible Descriptor with rich set of features to support buffer interrupt moderation, multiple transfers, isochronous, control, and scattered buffering support
- USB Class-Specific Device Features
 - 1. Stream support for UASP application
 - 2. Gathering of scattered packet to support Ethernet Over USB
 - 3. Scheduling of multiple Ethernet packets without interrupt
 - 4. Variable FIFO buffer allocation for each endpoint
 - 5. For isochronous applications, scheduling of variable-length payloads for each microframe
 - 6. Microframe precise scheduling for isochronous applications
 - 7. Configurable endpoint type selection and dynamic FIFO allocation to facilitate multi-function/composite device implementation. During set-config or alternate-setting, device resources are reconfigured to meet the configuration or alternate setting

requirements.

- USB 3.0 xHCI Host Features
 1. Support up to 64 devices
 2. Support 1 interrupter
 3. Support 1 USB2.0 port and 1 Super-Speed port
 4. Concurrent USB3.0/USB2.0 traffic, up to 8.48Gbps bandwidth
 5. Support standard or open-source xHCI and class driver
- USB 3.0 Dual-Role Device (DRD) Features
 1. Static Device Operation
 2. Static Host Operation
 3. USB3.0/USB2.0 OTG A device and B device basing on ID
 4. Not support USB3.0/USB2.0 OTG session request protocol(SRP), host negotiation protocol(HNP) and Role Swap Protocol(RSP)

9.2 Block Diagram

USB3.0 OTG Controller comprises with:

- Bus Interface/List Management: Register Interface/Data and Descriptors DMA management
- HS/FS/LS MAC : USB2.0 part logic
- SS MAC : SS part logic
- USB2.0 PHY: UTMI+ interface USB2.0 PHY
- PCIe&SS PHY: Pipe Interface PCIe&Super-Speed PHY

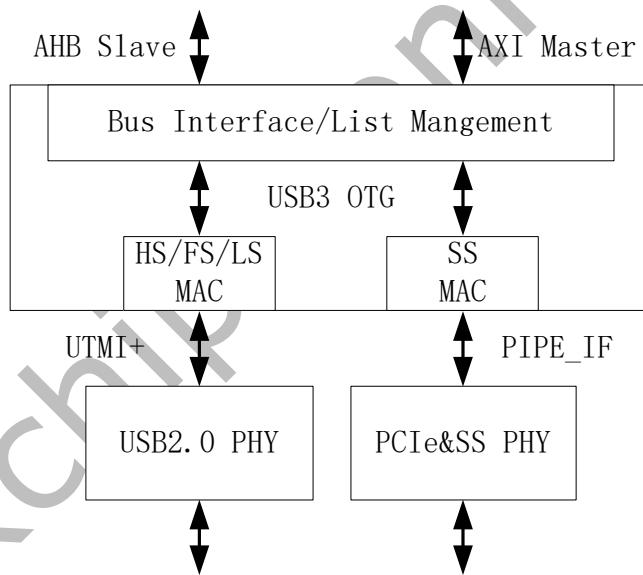


Fig. 9-1 USB3.0 OTG Block Diagram

9.3 Function Description

As a USB3.0 OTG controller, it can act as static xHCI host controller, static device controller, USB3.0/2.0 OTG A device or B device basing on ID of USB2.0 PHY.

As device controller, it can work on either USB2.0 speed or Super-Speed basing on speed of host attached to, and process USB tractions described in the descriptors (read back from external memory by AXI master) to/from UTMI+ interface and Pipe Interface of PCIE&SS PHY.

As host controller, it can work on USB2.0 speed, Super-Speed or both basing on speed or type of attached device, and process USB tractions described in the descriptors (read back from external memory by AXI master) to/from UTMI+ interface and Pipe Interface of PCIE&SS PHY.

9.4 Register Description

9.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

Table 9-1 USB3 Address Mapping

| Offset Address Range | Register Type |
|----------------------|--------------------------------------|
| 0x00000 ~ 0x07FFF | xHCI Registers, see xHCI spec. |
| 0x0C100 ~ 0x0C6FF | Global Registers |
| 0x0C700 ~ 0x0CBFF | Device Controller Registers |
| 0x0CC00 ~ 0x0CFFF | Unused/Reserved |
| 0x40000 ~ 0x7FFFF | Internal RAM0 – Debug Access (256KB) |
| 0x80000 ~ 0xBFFFF | Internal RAM1 – Debug Access (256KB) |
| 0xC0000 ~ 0xFFFFF | Internal RAM2 – Debug Access (256KB) |

9.4.2 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|--------------------|--------|------|-------------|---|
| USB3_GSBUSCFG0 | 0xc100 | W | 0x00000001 | Global SoC Bus Configuration Register 0 |
| USB3_GSBUSCFG1 | 0xc104 | W | 0x00000300 | Global SoC Bus Configuration Register 1 |
| USB3_GTXTHRCFG | 0xc108 | W | 0x00000000 | Global Tx Threshold Control Register |
| USB3_GRXTHRCFG | 0xc10c | W | 0x00000000 | Global Rx Threshold Control Register |
| USB3_GCTL | 0xc110 | W | 0x30c12004 | Global Core Control Register |
| USB3_GPMSTS | 0xc114 | W | 0x00000000 | Global Power Management Status Register |
| USB3_GSTS | 0xc118 | W | 0x7e800000 | Global Status Register |
| USB3_GUCTL1 | 0xc11c | W | 0x0004018a | Global User Control Register 1 |
| USB3_GSNPSID | 0xc120 | W | 0x5533300a | Global SNPS ID Register |
| USB3_GGPIO | 0xc124 | W | 0x0000abcd | Global General Purpose Input/Output Register |
| USB3_GUID | 0xc128 | W | 0x20180101 | Global User ID Register |
| USB3_GUCTL | 0xc12c | W | 0x02000010 | Global User Control Register |
| USB3_GBUSERRADDRLO | 0xc130 | W | 0x00000000 | Global SoC Bus Error Address Register - Low |
| USB3_GBUSERRADDRHI | 0xc134 | W | 0x00000000 | Global SoC Bus Error Address Register - High |
| USB3_GPRTBIMAPLO | 0xc138 | W | 0x00000000 | Global SS Port to Bus Instance Mapping Register - Low |
| USB3_GHWPARAMS0 | 0xc140 | W | 0x2020400a | Global Hardware Parameters Register 0 |
| USB3_GHWPARAMS1 | 0xc144 | W | 0x0160c93b | Global Hardware Parameters Register 1 |

| Name | Offset | Size | Reset Value | Description |
|---------------------|---------------|------|-------------|---|
| USB3_GHWPARAMS2 | 0xc148 | W | 0x20180101 | Global Hardware Parameters Register 2 |
| USB3_GHWPARAMS3 | 0xc14c | W | 0x08290085 | Global Hardware Parameters Register 3 |
| USB3_GHWPARAMS4 | 0xc150 | W | 0x47822010 | Global Hardware Parameters Register 4 |
| USB3_GHWPARAMS5 | 0xc154 | W | 0x04204108 | Global Hardware Parameters Register 5 |
| USB3_GHWPARAMS6 | 0xc158 | W | 0x075e8020 | Global Hardware Parameters Register 6 |
| USB3_GHWPARAMS7 | 0xc15c | W | 0x03880800 | Global Hardware Parameters Register 7 |
| USB3_GDBGFIFOSPACE | 0xc160 | W | 0x00420000 | Global Debug Queue/FIFO Space Available Register |
| USB3_GDBGLTSSM | 0xc164 | W | 0x41052840 | Global Debug LTSSM Register |
| USB3_GDBGLNMCC | 0xc168 | W | 0x00000000 | Global Debug LNMCC Register |
| USB3_GDBGBMU | 0xc16c | W | 0x00000000 | Global Debug BMU Register |
| USB3_GDBGLSPMUX | 0xc170 | W | 0x003f0000 | Global Debug LSP MUX Register - Device |
| USB3_GDBGLSP | 0xc174 | W | 0x00000000 | Global Debug LSP Register |
| USB3_GDBGEPIFO0 | 0xc178 | W | 0x00000000 | Global Debug Endpoint Information Register 0 |
| USB3_GDBGEPIFO1 | 0xc17c | W | 0x00800000 | Global Debug Endpoint Information Register 1 |
| USB3_GPRTBIMAP_HSL0 | 0xc180 | W | 0x00000000 | Global High-Speed Port to Bus Instance Mapping Register - Low |
| USB3_GPRTBIMAP_FSL0 | 0xc188 | W | 0x00000000 | Global Full-Speed Port to Bus Instance Mapping Register - Low |
| USB3_GUSB2PHYCFG0 | 0xc200 | W | 0x40102440 | Global USB2 PHY Configuration Register 0 |
| USB3_GUSB3PIPECTL0 | 0xc2c0 | W | 0x010e0002 | Global USB3 PIPE Control Register 0 |
| USB3_GTXFIFOSIZ0~6 | 0xc300~0xc324 | W | 0x00000042 | Global Transmit FIFO Size Register n |
| USB3_GRXFIFOSIZ0~2 | 0xc380~0xC388 | W | 0x00000285 | Global Receive FIFO Size Register n |
| USB3_GEVNTADRLO0 | 0xc400 | W | 0x00000000 | Global Event Buffer Address (Low) Register 0 |
| USB3_GEVNTADRHI0 | 0xc404 | W | 0x00000000 | Global Event Buffer Address (High) Register 0 |
| USB3_GEVNTSIZ0 | 0xc408 | W | 0x00000000 | Global Event Buffer Size Register 0 |

| Name | Offset | Size | Reset Value | Description |
|--------------------|---------------|------|-------------|---|
| USB3_GEVNTCOUNT0 | 0xc40c | W | 0x00000000 | Global Event Buffer Count Register 0 |
| USB3_GHWPARAMS8 | 0xc600 | W | 0x0000075e | Global Hardware Parameters Register 8 |
| USB3_GTXFIFOPRIDEV | 0xc610 | W | 0x00000000 | Global Device TX FIFO DMA Priority Register |
| USB3_GTXFIFOPRIHST | 0xc618 | W | 0x00000000 | Global Host TX FIFO DMA Priority Register |
| USB3_GRXFIFOPRIHST | 0xc61c | W | 0x00000000 | Global Host RX FIFO DMA Priority Register |
| USB3_GFIFOPRIDLBC | 0xc620 | W | 0x00000000 | Global Host Debug Capability DMA Priority Register |
| USB3_GDMAHLRATIO | 0xc624 | W | 0x00000080 | Global Host FIFO DMA High-Low Priority Ratio Register |
| USB3_GFLADJ | 0xc630 | W | 0x00000000 | Global Frame Length Adjustment Register |
| USB3_DCFG | 0xc700 | W | 0x00080804 | Device Configuration Register |
| USB3_DCTL | 0xc704 | W | 0x00f00000 | Device Control Register |
| USB3_DEVTEN | 0xc708 | W | 0x00000000 | Device Event Enable Register |
| USB3_DSTS | 0xc70c | W | 0x00d20004 | Device Status Register |
| USB3_DGCMDPAR | 0xc710 | W | 0x00000000 | Device Generic Command Parameter Register |
| USB3_DGCMD | 0xc714 | W | 0x00000000 | Device Generic Command Register |
| USB3_DALEPENA | 0xc720 | W | 0x00000000 | Device Active USB Endpoint Enable Register |
| USB3_DEPnCMDPAR2 | 0xc800~0xc8c0 | W | 0x00000000 | Device Physical Endpoint-n Command Parameter 2 Register |
| USB3_DEPnCMDPAR1 | 0xc804~0xc8c4 | W | 0x00000000 | Device Physical Endpoint-n Command Parameter 1 Register |
| USB3_DEPnCMDPAR0 | 0xc808~0xc8c8 | W | 0x00000000 | Device Physical Endpoint-n Command Parameter 0 Register |
| USB3_DEPnCMD | 0xc80c~0xc8cc | W | 0x00000000 | Device Physical Endpoint-n Command Register |

Notes:Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

9.4.3 Detail Register Description

USB3_GSBUSCFG0

Address: Operational Base + offset (0xc100)

Global SoC Bus Configuration Register 0

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:28 | RW | 0x0 | DATRDREQINFO AXI-cache for Data Read (DatRdReqInfo) |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 27:24 | RW | 0x0 | DESRDREQINFO AXI-cache for Descriptor Read (DesRdReqInfo). |
| 23:20 | RW | 0x0 | DATWRREQINFO AXI-cache for Data Write (DatWrReqInfo). |
| 19:16 | RW | 0x0 | DESWRREQINFO AXI-cache for Descriptor Write (DesWrReqInfo) |
| 15:12 | RO | 0x0 | reserved |
| 11 | RW | 0x0 | DATBIGEND This bit controls the endian mode for data accesses. 0, Little-endian (default); 1, Big-endian; |
| 10 | RW | 0x0 | DESBIGEND This bit controls the endian mode for descriptor accesses. 0, Little-endian (default); 1, Big-endian. |
| 9:8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | INCR256BRSTENA If software set this bit to 1, the AXI master uses INCR to do the 256-beat burst. |
| 6 | RW | 0x0 | INCR128BRSTENA If software set this bit to 1, the AXI master uses INCR to do the 128-beat burst. |
| 5 | RW | 0x0 | INCR64BRSTENA If software set this bit to 1, AXI master uses INCR to do the 64-beat burst. |
| 4 | RW | 0x0 | INCR32BRSTENA If software set this bit to 1, the AXI master uses INCR to do the 32-beat burst. |
| 3 | RW | 0x0 | INCR16BRSTENA If software set this bit to 1, the AXI master uses INCR to do the 16-beat burst. |
| 2 | RW | 0x0 | INCR8BRSTENA If software set this bit to 1, the AXI master uses INCR to do the 8-beat burst. |
| 1 | RW | 0x0 | INCR4BRSTENA When this bit is enabled the controller is allowed to do bursts of beat length 1, 2, 3, and 4. It is highly recommended that this bit is enabled to prevent descriptor reads and writes from being broken up into separate transfers. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RW | 0x1 | <p>INCRBRSTENA</p> <p>This bit determines the set of burst lengths the master interface uses. It works in conjunction with the GSBUSCFG0[7:1] enables (INCR256/128/64/32/16/8/4).</p> <p>0: INCRX burst mode ARLEN/AWLEN do not use INCR. They use only the following burst lengths: 1; 4 (if GSBUSCFG0.INCR4BrstEna = 1); 8 (if GSBUSCFG0.INCR8BrstEna = 1); 16 (if GSBUSCFG0.INCR16BrstEna = 1); 32 (if GSBUSCFG0.INCR32BrstEna = 1); 64 (if GSBUSCFG0.INCR64BrstEna = 1); 128 (if GSBUSCFG0.INCR128BrstEna = 1); 256 (if GSBUSCFG0.INCR256BrstEna = 1); 1: INCR (undefined length) burst mode ARLEN/AWLEN uses any length less than or equal to the largest-enabled burst length of INCR4/8/16/32/64/128/256. For cache line-aligned applications, this bit is typically set to 0 to ensure that the master interface uses only power-of-2 burst lengths (as enabled via GSBUSCFG0[7:0]).</p> |

USB3_GSBUSCFG1

Address: Operational Base + offset (0xc104)

Global SoC Bus Configuration Register 1

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:13 | RO | 0x0 | reserved |
| 12 | RW | 0x0 | <p>EN1KPAGE</p> <p>1K Page Boundary Enable</p> <p>By default (this bit is disabled) the AXI breaks transfers at the 4k page boundary. When this bit is enabled, the AXI master (DMA data) breaks transfers at the 1k page boundary.</p> |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 11:8 | RW | 0x3 | <p>PipeTransLimit AXI Pipelined Transfers Burst Request Limit The field controls the number of outstanding pipelined transfer requests the AXI master pushes to the AXI slave. When the AXI master reaches this limit, it does not make any more requests on the AXI ARADDR and AWADDR buses until the associated data phases complete.</p> <p>This field is encoded as follows:</p> <ul style="list-style-type: none"> 0: 1 request 1: 2 requests 2: 3 requests 3: 4 requests ... F: 16 requests |
| 7:0 | RO | 0x0 | reserved |

USB3_GTXTHRCFG

Address: Operational Base + offset (0xc108)

Global Tx Threshold Control Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29 | RO | 0x0 | <p>USBTxPktCntSel USB Transmit Packet Count Enable This field enables/disables the USB transmission multi-packet thresholding: 0: USB transmission multi-packet thresholding is disabled; the core can only start transmission on the USB after the entire packet has been fetched into the corresponding TXFIFO. 1: USB transmission multi-packet thresholding is enabled. The core can only start transmission on the USB after USB Transmit Packet Count amount of packets for the USB transaction (burst) are already in the corresponding TXFIFO. This mode is only valid in the host mode. It is only used for SuperSpeed.</p> |
| 28 | RO | 0x0 | reserved |
| 27:24 | RW | 0x0 | <p>USBTxPktCnt USB Transmit Packet Count This field specifies the number of packets that must be in the TXFIFO before the core can start transmission for the corresponding USB transaction (burst). This field is only valid when the USB Transmit Packet Count Enable field is set to one. Valid values are from 1 to 15. Note: This field must be less than or equal to the USB Maximum TX Burst Size field.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 23:16 | RW | 0x00 | <p>USBMaxTxBurstSize USB Maximum TX Burst Size When USBTxPktCntSel is 1, this field specifies the Maximum Bulk OUT burst the core can execute. When the system bus is slower than the USB, TX FIFO can underrun during a long burst. You can program a smaller value to this field to limit the TX burst size that the core can execute. It only applies to SS Bulk, Isochronous, and Interrupt OUT endpoints in the host mode. Valid values are from 1 to 16.</p> |
| 15:0 | RO | 0x0 | reserved |

USB3_GRXTHRCFG

Address: Operational Base + offset (0xc10c)

Global Rx Threshold Control Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29 | RW | 0x0 | <p>USBRxPktCntSel USB ReceivePacket Count Enable This field enables/disables the USB reception multi-packet thresholding: 0: The core can only start reception on the USB when the RX FIFO has space for at least one packet. 1: The core can only start reception on the USB when the RX FIFO has space for at least USBRxPktCnt amount of packets. This mode is valid in both host and device mode. It is only used for SuperSpeed. In device mode, Setting this bit to 1 also enables the functionality of reporting NUMP in the ACK TP based on the RX FIFO space instead of reporting a fixed NUMP derived from DCFG.NUMP If you are using external buffer control (EBC) feature, disable this mode by setting USBRxPktCntSel to 0.</p> |
| 28 | RO | 0x0 | reserved |
| 27:24 | RW | 0x0 | <p>USBRxPktCnt USB Receive Packet Count In host mode, this field specifies the space (in terms of the number of packets) that must be available in the RX FIFO before the core can start the corresponding USB RX transaction (burst). In device mode, this field specifies the space (in terms of the number of packets) that must be available in the RX FIFO before the core can send ERDY for a flow-controlled endpoint. This field is valid only when the USB Receive Packet Count Enable field is set to 1. The valid values for this field are from 1 to 15. Note: This field must be less than or equal to the USB Maximum Receive Burst Size field.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 23:19 | RW | 0x00 | <p>USBMaxRxBurstSize USB Maximum Receive Burst Size In host mode, this field specifies the Maximum Bulk IN burst the usb3 controller can perform. When the system bus is slower than the USB, RX FIFO can overrun during a long burst.</p> <p>You can program a smaller value to this field to limit the RX burst size that the core can perform. It only applies to SS Bulk, Isochronous, and Interrupt IN endpoints in the host mode.</p> <p>In device mode, this field specifies the NUMP value that is sent in ERDY for an OUT endpoint.</p> <p>This field is valid only when USBRxPktCntSel is one. The valid values for this field are from 1 to 16.</p> |
| 18:0 | RO | 0x0 | reserved |

USB3_GCTL

Address: Operational Base + offset (0xc110)
Global Core Control Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:19 | RW | 0x0618 | <p>PWRDNSCALE Power Down Scale (PwrDnScale) The USB3 suspend_clk input replaces pipe3_rx_pclk as a clock source to a small part of the USB3 core that operates when the SS PHY is in its lowest power (P3) state, and therefore does not provide a clock.</p> <p>The Power Down Scale field specifies how many suspend_clk periods fit into a 16 kHz clock period. When performing the division, round up the remainder.</p> <p>For example, when using an 8-bit/16-bit/32-bit PHY and 25-MHz Suspend clock,</p> $\text{Power Down Scale} = 25000 \text{ kHz}/16 \text{ kHz} = 13'd1563 \text{ (rounder up)}$ <p>Note:</p> <p>Minimum Suspend clock frequency is 32 kHz</p> <p>Maximum Suspend clock frequency is 125 MHz</p> <p>The LTSSM uses Suspend clock for 12-ms and 100-ms timers during suspend mode. According to the USB 3.0 specification, the accuracy on these timers is 0% to +50%.</p> <p>$12 \text{ ms} + 0\text{~}+50\% \text{ accuracy} = 18 \text{ ms}$ (Range is 12 ms - 18 ms)</p> <p>$100 \text{ ms} + 0\text{~}+50\% \text{ accuracy} = 150 \text{ ms}$ (Range is 100 ms - 150ms).</p> <p>The suspend clock accuracy requirement is:</p> $(12,000/62.5) * (\text{GCTL}[31:19]) * \text{actual suspend_clk_period}$ <p>must be between 12,000 and 18,000</p> $(100,000/62.5) * (\text{GCTL}[31:19]) * \text{actual suspend_clk_period}$ <p>must be between 100,000 and 150,000</p> <p>For example, if your suspend_clk frequency varies from 7.5 MHz to 10.5MHz, then the value needs to programmed is:</p> <p>Power Down Scale = $10500/16 = 657$ (rounded up; and fastest frequency used).</p> |
| 18 | RW | 0x0 | <p>MASTERFILTBYPASS Master Filter Bypass</p> <p>When this bit is set to 1'b1, all the filters are bypassed. The double synchronizers to mac_clk preceding the filters are also bypassed. For enabling the filters, this bit must be 1'b0.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17 | RW | 0x0 | <p>BYPSETADDR Bypass SetAddress in Device Mode. When BYPSSETADDR bit is set, the device core uses the value in the DCFG[DevAddr] bits directly for comparing the device address in the tokens.</p> <p>For simulation, you can use this feature to avoid sending an actual SET ADDRESS control transfer on the USB, and make the device core respond to a new address.</p> <p>When the xHCI Debug capability is enabled and this bit is set, the Debug Target immediately enters the configured state without requiring the Debug Host to send a SetAddress or SetConfig request.</p> <p>Note: You can set this bit for simulation purposes only. In the actual hardware, this bit must be set to 1'b0.</p> |
| 16 | RW | 0x1 | <p>U2RSTECN U2RSTECN If the SuperSpeed connection fails during POLL or LMP exchange, the device connects at non-SS mode.</p> <p>If this bit is set, then device attempts three more times to connect at SS, even if it previously failed to operate in SS mode. For each attempt, the device checks receiver termination eight times.</p> <p>From 2.60a release, this bit controls whether to check for Rx.Detect eight times or one time for every attempt. Device controller on USB 2.0 reset checks for receiver termination eight times per attempt if this bit is set to zero, or only once per attempt if the bit is set to one.</p> <p>Note: This bit is applicable only in device mode.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 15:14 | RW | 0x0 | <p>FRMSCLDWN FRMSCLDWN</p> <p>This field scales down device view of a SOF/USOF/ITP duration.</p> <p>For SS/HS mode:</p> <ul style="list-style-type: none"> Value of 2'h3 implements interval to be 15.625 us Value of 2'h2 implements interval to be 31.25 us Value of 2'h1 implements interval to be 62.5 us Value of 2'h0 implements interval to be 125us <p>For FS mode, the scale-down value is multiplied by 8.</p> <p>When xHCI Debug Capability is enabled, this field also scales down the MaxPacketSize of the IN and OUT bulk endpoint to allow more traffic during simulation. It can only be changed from a non-zero value during simulation.</p> <p>2'h0: 1024 bytes 2'h1: 512 bytes 2'h2: 256 bytes 2'h3: 128 bytes</p> |
| 13:12 | RW | 0x2 | <p>PRTCAPDIR</p> <p>PRTCAPDIR: Port Capability Direction (PrtCapDir)</p> <p>2'b01: for Host configurations 2'b10: for Device configurations</p> <p>SW should base on IDDIG input to set usb3 controller as an OTG 2.0/3.0 device with A-device or B-device.</p> |
| 11 | RW | 0x0 | <p>CORESOFTRESET</p> <p>Core Soft Reset (CoreSoftReset)</p> <p>1'b0 - No soft reset; 1'b1 - Soft reset to core</p> <p>Clears the interrupts and all the CSRs except the following registers:</p> <p>GCTL; GUCTL; GSTS; GSNPSSID; GPIO; GUID; GUSB2PHYCFGn registers; GUSB3PIPECTLn registers; DCFG; DCTL; DEVTN; DSTS.</p> <p>When you reset PHYs (using GUBS3PHYCFG or GUSB3PIPECTL registers), you must keep the core in reset state until PHY clocks are stable. This controls the bus, ram, and mac domain resets.</p> <p>Note: This bit is for debug purposes only. Use USBCMD.HCRESET in xHCI Mode and DCTL.SoftReset in device mode for soft reset.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 10 | RW | 0x0 | <p>SOFITPSYNC SOFITPSYNC</p> <p>If this bit is set to 0 operating in host mode, the core keeps the UTMI/ULPI PHY on the first port in a non-suspended state whenever there is a SuperSpeed port that is not in Rx.Detect, SS.Disable and U3.</p> <p>If this bit is set to 1 operating in host mode, the core keeps the UTMI/ULPI PHY on the first port in a non-suspended state whenever the other non-SuperSpeed ports are not in a suspended state. This feature is useful because it saves power by suspending UTMI/ULPI when SuperSpeed only is active, and it helps resolve when the PHY does not transmit a host resume unless it is placed in suspend state. This bit must be programmed as a part of initialization at power-on reset, and must not be dynamically changed afterwards.</p> <p>Note:</p> <p>USB2PHYCFGn[6].PhySusp eventually decides to put the UTMI/ULPI PHY in to suspend state. In addition, when this bit is set to 1, the core generates ITP from the ref_clk based counter. Otherwise, ITP and SOF are generated from utmi/ulpi_clk[0] based counter. To program the reference clock period inside the core, refer to GUCTL[31:22].REFCLKPER.</p> <p>This feature is valid in Host and DRD/OTG configurations and used only in Host mode operation.</p> <p>If you never use this feature or the GFLADJ.GFLADJ_REFCLK_LPM_SEL, the minimum frequency for the ref_clk can be as low as 32KHz. You can connect the suspend_clk (as low as 32 KHz) to the ref_clk.</p> <p>If you plan to enable hardware-based LPM or software-based LPM (PORTPMSC. HLE=1), then you cannot use this feature.</p> <p>Turn off this feature by setting this bit to 0 and use the GFLADJ.GFLADJ_REFCLK_LPM_SEL feature.</p> <p>If you set this bit to 1, the GUSB2PHYCFG.U2_FREECLK_EXISTS bit and the DWC_USB3_FREECLK_USB2_EXIST parameter must be set to 0.</p> <p>Program this bit to 0 if the core is intended to be operated in USB 3.0 mode.</p> |
| 9 | RW | 0x0 | <p>U1U2TimerScale</p> <p>Disable U1/U2 timer Scaledown (U1U2TimerScale).</p> <p>If set to 1 along with GCTL[5:4] (ScaleDown) = 2'bX1, disables the scale down of U1/U2 inactive timer values. This is for simulation mode only.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 8 | RW | 0x0 | <p>DEBUGATTACH Debug Attach When this bit is set:</p> <ol style="list-style-type: none"> 1. SS Link proceeds directly to the Polling link state (after RUN/STOP in the DCTL register is asserted) without checking remote termination; 2. Link LFPS polling timeout is infinite; 3. Polling timeout during TS1 is infinite (in case link is waiting for TXEQ to finish). |
| 7:6 | RW | 0x0 | <p>RAMCLKSEL RAM Clock Select (RAMClkSel) 2'b00: bus clock 2'b01: pipe clock (Only used in device mode) 2'b10: In device mode, pipe/2 clock. In Host mode, controller switches ram_clk between pipe/2 clock, mac2_clk and bus_clk based on the status of the U2/U3 ports 2'b11: In device mode, selects mac2_clk as ram_clk (when 8-bit UTMI or ULPI used. Not supported in 16-bit UTMI mode); In Host mode, controller switches ram_clk between pipe_clk, mac2_clk and bus_clk based on the status of the U2/U3 ports. In device mode, upon a USB reset and USB disconnect, the hardware clears these bits to 2'b00.</p> |
| 5:4 | RW | 0x0 | <p>SCALEDOWN Scale-Down Mode (ScaleDown) When Scale-Down mode is enabled for simulation, the core uses scaled-down timing values, resulting in faster simulations. When Scale-Down mode is disabled, actual timing values are used. This is required for hardware operation. HS/FS/LS Modes: 2'b00: Disables all scale-downs. Actual timing values are used. 2'b01: Enables scale-down of all timing values except Device mode suspend and resume. These include Speed enumeration, HNP/SRP, and Host mode suspend and resume 2'b10: Enables scale-down of Device mode suspend and resume timing values only. 2'b11: Enables bit 0 and bit 1 scale-down timing values. SS Mode: 2'b00: Disables all scale-downs. Actual timing values are used. 2'b01: Enables scaled down SS timing and repeat values including: (1) Number of TxEq training sequences reduce to 8; (2) LFPS polling burst time reduce to 256 nS; (3) LFPS warm reset receive reduce to 30 uS. 2'b10: No TxEq training sequences are sent. Overrides Bit 4. 2'b11: Enables bit 0 and bit 1 scale-down timing values.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 3 | RW | 0x0 | DISSCRAMBLE Disable Scrambling (DisScramble) Transmit request to Link Partner on next transition to Recovery or Polling. |
| 2 | RW | 0x1 | U2EXIT_LFPS U2EXIT_LFPS If this bit is: 0: the link treats 248ns LFPS as a valid U2 exit. 1: the link waits for 8us of LFPS before it detects a valid U2 exit. This bit is added to improve interoperability with a third party host controller. This host controller in U2 state while performing receiver detection generates an LFPS glitch of about 4ms duration. This causes the device to exit from U2 state because the LFPS filter value is 248ns. With the new functionality enabled, the device can stay in U2 while ignoring this glitch from the host controller. |
| 1 | RO | 0x0 | GblHibernationEn GblHibernationEn This bit enables hibernation at the global level. If hibernation is not enabled through this bit, the PMU immediately accepts the D0->D3 and D3->D0 power state change requests, but does not save or restore any core state. In addition, the PMUs never drive the PHY interfaces and let the core continue to drive the PHY interfaces. |
| 0 | RW | 0x0 | DSBLCLKGTNG Disable Clock Gating (DsblClkGtng) This bit is set to 1 and the core is in Low Power mode, internal clock gating is disabled. You can set this bit to 1'b1 after Power On Reset. |

USB3_GPMSTS

Address: Operational Base + offset (0xc114)

Global Power Management Status Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:28 | WO | 0x0 | PortSel Global Power Management Status Register This field selects the port number. |
| 27:17 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 16:12 | RO | 0x00 | <p>U3Wakeup U3Wakeup</p> <p>This field gives the following USB 3.0 port wakeup conditions:</p> <ul style="list-style-type: none"> Bit [12]: Overcurrent Detected Bit [13]: Resume Detected Bit [14]: Connect Detected Bit [15]: Disconnect Detected Bit [16]: Last Connection State |
| 11:10 | RO | 0x0 | reserved |
| 9:0 | RO | 0x000 | <p>U2Wakeup U2Wakeup</p> <p>This field indicates the following USB 2.0 port wakeup conditions:</p> <ul style="list-style-type: none"> Bit [0]: Overcurrent Detected Bit [1]: Resume Detected Bit [2]: Connect Detected Bit [3]: Disconnect Detected Bit [4]: Last Connection State Bit [5]: ID Change Detected Bit [6]: SRP Request Detected Bit [7]: ULPI Interrupt Detected Bit [8]: USB Reset Detected Bit [9]: Resume Detected Changed |

USB3_GSTS

Address: Operational Base + offset (0xc118)

Global Status Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:20 | RO | 0x7e8 | <p>CBELT Current BELT Value</p> <p>In Host mode, this field indicates the minimum value of all received device BELT values and the BELT value that is set by the Set Latency Tolerance Value command.</p> |
| 19:12 | RO | 0x0 | reserved |
| 11 | RO | 0x0 | <p>SSIC_IP SSIC interrupt pending</p> <p>This field indicates that there is a pending interrupt related to SSIC in the SEVT register.</p> <p>Note: When the DWC_USB3_NUM_SSIC_PORTS parameter is set to zero, this bit is reserved.</p> |
| 10 | RO | 0x0 | <p>OTG_IP OTG Interrupt Pending</p> <p>This field indicates that there is a pending interrupt pertaining to OTG in OEVT register.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 9 | RO | 0x0 | BC_IP Battery Charger Interrupt Pending This field indicates that there is a pending interrupt pertaining to BC in BCEVT register. |
| 8 | RO | 0x0 | ADP_IP ADP Interrupt Pending his field indicates that there is a pending interrupt pertaining to ADP in ADPEVT register. |
| 7 | RO | 0x0 | Host_IP Host Interrupt Pending This field indicates that there is a pending interrupt pertaining to xHC in the Host event queue. |
| 6 | RO | 0x0 | Device_IP Device Interrupt Pending This field indicates that there is a pending interrupt pertaining to peripheral (device) operation in the Device event queue. |
| 5 | W1C | 0x0 | CSRTimeout CSR Timeout When this bit is 1'b1, it indicates that the software performed a write or read to a core register that could not be completed within DWC_USB3_CSR_ACCESS_TIMEOUT bus clock cycles (default: h1FFFF). |
| 4 | W1C | 0x0 | BUSERRADDRVLD Bus Error Address Valid Indicates that the GBUSERRADDR register is valid and reports the first bus address that encounters a bus error. |
| 3:2 | RO | 0x0 | reserved |
| 1:0 | RO | 0x0 | CURMOD Current Mode of Operation Current Mode of Operation |

USB3_GUCTL1

Address: Operational Base + offset (0xc11c)

Global User Control Register 1

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:30 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 29 | RW | 0x0 | <p>FILTER_SE0_FSLS_EOP FILTER_SE0_FSLS_EOP</p> <p>0: Default behaviour, no change in Linestate check for SE0 detection in FS/LS</p> <p>1: Feature enabled, FS/LS SE0 is filtered for 2 clocks for detecting EOP</p> <p>This bit is applicable for FS/LS operation. If this feature is enabled, then SE0 on the linestate is validated for 2 consecutive utmi/ulpi clock edges for EOP detection. This feature is applicable only in FS in device mode and FS/LS mode of operation in host mode.</p> <p>Device mode: FS - If GUCTL1.FILTER_SE0_FSLS_EOP is set, then for device LPM handshake, the core will ignore single SE0 glitch on the linestate during transmit. Only 2 or more SE0 is considered as a valid EOP on FS.</p> <p>Host mode: FS/LS - If GUCTL1.FILTER_SE0_FSLS_EOP is set, then the core will ignore single SE0 glitch on the linestate during transmit. Only 2 or more SE0 is considered as a valid EOP on FS/LS port. Enable this feature if the LineState has SE0 glitches during transmission. This bit is quasi-static, i.e., should not be changed during device operation.</p> |
| 28 | RW | 0x0 | <p>TX_IPGAP_LINECHECK_DIS TX_IPGAP_LINECHECK_DIS</p> <p>0: Default behaviour, no change in Linestate check</p> <p>1: Feature enabled, 2.0 MAC disables Linestate check during HS transmit</p> <p>This bit is applicable for HS operation of u2mac. If this feature is enabled, then the 2.0 mac operating in HS ignores the UTMI/ULPI Linestate during the transmit of a token (during token-to-token and token-to-data IPGAP). When enabled, the controller implements a fixed 40-bit TxEndDelay after the packet is given on UTMI and ignores the Linestate during this time. This feature is applicable only in HS mode of operation.</p> <p>Device mode: If GUCTL1.TX_IPGAP_LINECHECK_DIS is set, then for device LPM handshake, the core will ignore the linestate after TX and wait for a fixed clocks (40 bit times equivalent) after transmitting ACK on utmi.</p> <p>Host mode: If GUCTL1.TX_IPGAP_LINECHECK_DIS is set, then the ipgap between (tkn to tkn/data) is added by 40 bit times of TXENDDELAY, and linestate is ignored during this 40 bit times delay.</p> <p>Enable this bit if the LineState will not reflect the expected line state (J) during transmission. This bit is quasi-static, i.e., should not be changed during device operation.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 27 | RW | 0x0 | <p>DEV_TRB_OUT_SPR_IND DEV_TRB_OUT_SPR_IND</p> <p>0: Default behaviour, no change in TRB status dword 1: Feature enabled, OUT TRB status indicates Short Packet</p> <p>This bit is applicable for device mode only (and ignored in host mode). If the device application (SW/HW) wants to know if a short packet was received for an OUT in the TRB status itself, then this feature can be enabled, so that a bit is set in the TRB writeback in the buf_size dword. Bit[26] - SPR of the trbstatus, RSVD, SPR,PCM1, bufsiz dword will be set during an OUT transfer TRB write back if this is the last TRB used for that transfer descriptor. This bit is quasi-static, i.e., should not be changed during device operation.</p> |
| 26 | RW | 0x0 | <p>DEV_FORCE_20_CLK_FOR_30_CLK DEV_FORCE_20_CLK_FOR_30_CLK</p> <p>0: Default behaviour, Uses 3.0 clock when operating in 2.0 mode 1: Feature enabled</p> <p>This bit is applicable (and to be set) for device mode (DCFG.Speed!= SS) only. In the 3.0 device core, if the core is programmed to operate in 2.0 only (i.e., Device Speed is programmed to 2.0 speeds in DCFG[Speed]), then setting this bit makes the internal 2.0 (utmi/ulpi) clock to be routed as the 3.0 (pipe) clock. Enabling this feature allows the pipe3 clock to be not-running when forcibly operating in 2.0 device mode. Note: When using this feature, all pipe3 inputs must be in inactive mode, esp. pipe3 clocks not running and pipe3_phystatus_async must be tied to 0. This bit should not be set if the core is programmed to operate in SuperSpeed mode (even when it falls back to 2.0). This bit is quasi-static, i.e., should not be changed during operation.</p> |
| 25 | RW | 0x0 | <p>P3_IN_U2 P3_IN_U2</p> <p>0: Default behaviour, When SuperSpeed link is in U2 , PowerState P2 is attempted on the PIPE Interface. 1: When SuperSpeed link is in U2, PowerState P3 is attempted if GUSB3PIPECTL[17] is set.</p> <p>Setting this bit enables P3 Power State when the SuperSpeed link is in U2. Another Power Saving option. Check with your PHY vendor before enabling this option. When setting this bit to 1 to enable P3 in P2, GUSB3PIPECTL[27] should be set to 0 to make sure that the U2 exit is attempted in P0.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 24 | RW | 0x0 | <p>DEV_L1_EXIT_BY_HW DEV_L1_EXIT_BY_HW 0: Default behaviour, disables device L1 hardware exit logic 1: feature enabled</p> <p>This bit is applicable for device mode (2.0) only. This field enables device controller sending remote wakeup for L1 if the device becomes ready for sending/accepting data when in L1 state. If the host expects the device to send remote wkp signalling to resume after going into L1 in flow controlled state, then this bit can be set to send the remote wake signal automatically when the device controller becomes ready. This HW remote wake feature is applicable only to bulk and interrupt transfers, and not for Isoch/Control</p> <p>When control transfers are in progress, the LPM will be rejected (NYET response). Only after control transfers are completed (either with ACK/STALL), LPM will be accepted</p> <p>For Isoch transfers, the host needs to do the wake-up and start the transfer. Device controller will not do remote-wakeup when Isoch endpoints get ready. The device SW needs to keep the GUSB2PHYCFG[EnblSlpM] reset in order to keep the PHY clock to be running for keeping track of SOF intervals.</p> <p>When L1 hibernation is enabled, the controller will not do automatic exit for hibernation requests thru L1.</p> <p>This bit is quasi-static, i.e., should not be changed during device operation.</p> |
| 23:21 | RW | 0x0 | <p>IP_GAP_ADD_ON IP_GAP_ADD_ON</p> <p>This register field is used to add on to the default inter packet gap setting in the USB 2.0 MAC. This should be programmed to a non zero value only in case where you need to increase the default inter packet delay calculations in the USB 2.0 MAC module DWC_usb3_u2mac.v</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 20 | RW | 0x0 | <p>DEV_LSP_TAIL_LOCK_DIS DEV_LSP_TAIL_LOCK_DIS</p> <p>0: Default behaviour, enables device lsp lock logic for tail TRB update</p> <p>1: Fix disabled</p> <p>This is a bug fix for STAR 9000716195 that affects the CSP mode for OUT endpoints in device mode. The issue is that tail TRB index is not synchronized with the cache Scratchpad bytecount update. If the fast-forward request comes in-between the bytecount update on a newly fetched TRB and the tail-index write update in TPF, the RDP works on an incorrect tail index and misses the byte count decrement for the newly fetched TRB in the fast-forwarding process. This fix needs to be present all the times.</p> |
| 19 | RW | 0x0 | <p>NAK_PER_ENH_FS NAK_PER_ENH_FS</p> <p>1: Enables performance enhancement for FS async endpoints in the presence of NAKs</p> <p>0: Enhancement not applied</p> <p>If a periodic endpoint is present , and if a bulk endpoint which is also active is being NAKed by the device, then this could result in a decrease in performance of other Full Speed bulk endpoint which is ACKed by the device. Setting this bit to 1, will enable the host controller to schedule more transactions to the async endpoints (bulk/ control) and hence will improve the performance of the bulk endpoint. This control bit should be enabled only if the existing performance with the default setting is not sufficient for your FullSpeed application. Setting this bit will only control, and is only required for Full Speed transfers.</p> |
| 18 | RW | 0x1 | <p>NAK_PER_ENH_HS NAK_PER_ENH_HS</p> <p>1: Enables performance enhancement for HS async endpoints in the presence of NAKs</p> <p>0: Enhancement not applied</p> <p>If a periodic endpoint is present , and if a bulk endpoint which is also active is being NAKed by the device, then this could result in a decrease in performance of other High Speed bulk endpoint which is ACKed by the device. Setting this bit to 1, will enable the host controller to schedule more transactions to the async endpoints (bulk/ control) and hence will improve the performance of the bulk endpoint. This control bit should be enabled only if the existing performance with the default setting is not sufficient for your HighSpeed application. Setting this bit will only control, and is only required for High Speed transfers.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17 | RW | 0x0 | PARKMODE_DISABLE_SS PARKMODE_DISABLE_SS This bit is used only in host mode, and is for debug purpose only. When this bit is set to 1 all SS bus instances in park mode are disabled. |
| 16 | RW | 0x0 | PARKMODE_DISABLE_HS PARKMODE_DISABLE_HS This bit is used only in host mode. When this bit is set to 1 all HS bus instances park mode are disabled. To improve performance in park mode, the xHCI scheduler queues in three requests of 4 packets each for High Speed asynchronous endpoints in a micro-frame. But if a device is slow and if it NAKs more than 3 times, then it is rescheduled only in the next micro-frame. This could decrease the performance of a slow device even further. In a few high speed devices (such as Sandisk Cruzer Blade 4GB VID:1921, PID:21863 and Flex Drive VID:3744, PID:8552) when an IN request is sent within 900ns of the ACK of the previous packet, these devices send a NAK. When connected to these devices, if required, the software can disable the park mode if you see performance drop in your system. When park mode is disabled, pipelining of multiple packet is disabled and instead one packet at a time is requested by the scheduler. This allows up to 12 NAKs in a micro-frame and improves performance of these slow devices. |
| 15 | RW | 0x0 | PARKMODE_DISABLE_FSLS PARKMODE_DISABLE_FSLS This bit is used only in host mode, and is for debug purpose only. When this bit is set to 1 all FS/LS bus instances in park mode disabled. |
| 14:9 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 8 | RW | 0x1 | <p>L1_SUSP_THRLD_EN_FOR_HOST L1_SUSP_THRLD_EN_FOR_HOST</p> <p>This bit is used only in host mode.</p> <p>The host controller asserts the utmi_l1_suspend_n and utmi_sleep_n output signals (see LPM Interface Signals table in the Databook) as follows:</p> <p>The controller asserts the utmi_l1_suspend_n signal to put the PHY into deep low-power mode in L1 when both of the following are true:</p> <p>The HIRD/BESL value used is greater than or equal to the value in L1_SUSP_THRLD_FOR_HOST field.</p> <p>The L1_SUSP_THRLD_EN_FOR_HOST bit is set to 1'b1. The controller asserts utmi_sleep_n on L1 when one of the following is true:</p> <p>The HIRD/BESL value used is less than the value in L1_SUSP_THRLD_FOR_HOST field.</p> <p>The L1_SUSP_THRLD_EN_FOR_HOST bit is set to 1'b0.</p> |
| 7:4 | RW | 0x8 | <p>L1_SUSP_THRLD_FOR_HOST L1_SUSP_THRLD_FOR_HOST</p> <p>This field is effective only when the L1_SUSP_THRLD_EN_FOR_HOST bit is set to 1. For more details, refer to the description of the L1_SUSP_THRLD_EN_FOR_HOST bit.</p> |
| 3 | RW | 0x1 | <p>HC_ERRATA_ENABLE Host ELD Enable</p> <p>When this bit is set to 1, it enables the Exit Latency Delta (ELD) support defined in the xHCI 1.0 Errata.</p> <p>This bit is used only in the host mode. This bit has to be set to 1 in Host mode.</p> |
| 2 | RW | 0x0 | <p>HC_PARCHK_DISABLE Host Parameter Check Disable</p> <p>When this bit is set to 0 (by default), the xHC checks that the input slot/EP context fields comply to the xHCI Specification.</p> <p>Upon detection of a parameter error during command execution, the xHC generates an event TRB with completion code indicating PARAMETER ERROR.</p> <p>When the bit is set to 1, the xHC does not perform parameter checks and does not generate PARAMETER ERROR completion code.</p> |
| 1 | RW | 0x1 | <p>OVRLD_L1_SUSP_COM OVRLD_L1_SUSP_COM</p> <p>If this bit is set, the utmi_l1_suspend_com_n is overloaded with the utmi_sleep_n signal. This bit is usually set if the PHY stops the port clock during L1 sleep condition.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RW | 0x0 | LOA_FILTER_EN LOA_FILTER_EN If this bit is set, the USB 2.0 port babble is checked at least three consecutive times before the port is disabled. This prevents false triggering of the babble condition when using low quality cables. Note: This bit is valid only in host mode. |

USB3_GSNPSID

Address: Operational Base + offset (0xc120)

Global SNPS ID Register

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x5533300a | SNPSID SNPSID SNPSID[31:16] indicates Core Identification Number. 0x5533 is ASCII for U3 (DWC_usb3). SNPSID[15:0] indicates the release number. Current Release is 3.00a. Software uses this register to configure release-specific features in the driver. |

USB3_GGPIO

Address: Operational Base + offset (0xc124)

Global General Purpose Input/Output Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | GPO General Purpose Output This field's value is driven out on the gp_out[15:0] core output port. |
| 15:0 | RO | 0x0000 | GPI General Purpose Input This field's read value reflects the gp_in[15:0] core input value. |

USB3_GUID

Address: Operational Base + offset (0xc128)

Global User ID Register

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x12345678 | USERID USERID Application-programmable ID field. |

USB3_GUCTL

Address: Operational Base + offset (0xc12c)

Global User Control Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:22 | RW | 0x008 | <p>REFCLKPER REFCLKPER</p> <p>This field indicates in terms of nano seconds the period of ref_clk. The default value of this register is set to 'h8 (8ns/125 MHz). This field needs to be updated during power-on initialization, if GCTL.SOFITPSYNC or GFLADJ.GFLADJ_REFCLK_LPM_SEL is set to 1. The programmable maximum value is 62ns, and the minimum value is 8ns.</p> <p>You must use a reference clock with a period that is an integer multiple, so that ITP can meet the jitter margin of 32ns. The allowable ref_clk frequencies whose period is not integer multiples are 16/17/19.2/24/39.7MHz.</p> <p>This field must not be set to 0 at any time. If you never plan to use this feature, then set this field to 'h8, the default value.</p> |
| 21 | RW | 0x0 | <p>NoExtrDI No Extra Delay Between SOF and the First</p> <p>Some HS devices misbehave when the host sends a packet immediately after a SOF. However, adding an extra delay between a SOF and the first packet can reduce the USB data rate and performance.</p> <p>This bit is used to control whether the host must wait for 2 microseconds before it sends the first packet after a SOF, or not. User can set this bit to one to improve the performance if those problematic devices are not a concern in the user's host environment.</p> <p>1'b0: Host waits for 2 microseconds after a SOF before it sends the first USB packet.</p> <p>1'b1: Host doesn't wait after a SOF before it sends the first USB packet.</p> |
| 20:18 | RO | 0x0 | reserved |
| 17 | RW | 0x0 | <p>SprsCtrlTransEn Sparse Control Transaction Enable</p> <p>Some devices are slow in responding to Control transfers. Scheduling multiple transactions in one microframe/frame can cause these devices to misbehave.</p> <p>If this bit is set to 1'b1, the host controller schedules transactions for a Control transfer in different microframes/frames.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | RW | 0x0 | <p>ResBwHSEPS Reserving 85% Bandwidth for HS Periodic EPs By default, HC reserves 80% of the bandwidth for periodic EPs. If this bit is set, the bandwidth is relaxed to 85% to accommodate two high speed, high bandwidth ISOC EPs.</p> <p>USB 2.0 required 80% bandwidth allocated for ISOC traffic. If two High-bandwidth ISOC devices (HD Webcams) are connected, and if each requires 1024-bytes X 3 packets per Micro-Frame, then the bandwidth required is around 82%. If this bit is set, then it is possible to connect two Webcams of 1024bytes X 3 payload per Micro-Frame each. Otherwise, you may have to reduce the resolution of the Webcams.</p> <p>This bit is valid in Host and DRD configuration and is used in host mode operation only. Ignore this bit in device mode.</p> |
| 15 | RW | 0x1 | <p>CMdevAddr Compliance Mode for Device Address When this bit is 1'b1, Slot ID may have different value than Device Address if max_slot_enabled < 128.</p> <p>1'b1: Increment Device Address on each Address Device command.</p> <p>1'b0: Device Address is equal to Slot ID. The xHCI compliance requires this bit to be set to 1. The 0 mode is for debug purpose only. This allows you to easily identify a device connected to a port in the Lecroy or Eliisys trace during hardware debug.</p> <p>This bit is valid in Host and DRD configuration and is used in host mode operation only. Ignore this bit in device mode.</p> |
| 14 | RW | 0x0 | <p>USBHstInAutoRetryEn Host IN Auto Retry When set, this field enables the Auto Retry feature. For IN transfers (non-isochronous) that encounter data packets with CRC errors or internal overrun scenarios, the auto retry feature causes the Host core to reply to the device with a non-terminating retry ACK (that is, an ACK transaction packet with Retry = 1 and NumP != 0).</p> <p>If the Auto Retry feature is disabled (default), the core will respond with a terminating retry ACK (that is, an ACK transaction packet with Retry = 1 and NumP = 0).</p> <p>1'b0: Auto Retry Disabled 1'b1: Auto Retry Enabled Note: This bit is also applicable to the device mode.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 13 | RW | 0x0 | <p>EnOverlapChk Enable Check for LFPS Overlap During Remote Ux 1'b1: The SuperSpeed link when exiting U1/U2/U3 waits for either the remote link LFPS or TS1/TS2 training symbols before it confirms that the LFPS handshake is complete. This is done to handle the case where the LFPS glitch causes the link to start exiting from the low power state. Looking for the LFPS overlap makes sure that the link partner also sees the LFPS. 1'b0: When the link exists U1/U2/U3 because of a remote exit, it does not look for an LFPS overlap.</p> |
| 12 | RW | 0x0 | <p>ExtCapSupptEN External Extended Capability Support Enable When set, this field enables extended capabilities to be implemented outside the core. When the ExtCapSupEN is set and the Debug Capability is enabled, the Next Capability pointer in Debug Capability returns 16. A read to the first DWORD of the last internal extended capability (the "xHCI Supported Protocol Capability for USB 3.0" when the Debug Capability is not enabled) returns a value of 4 in the Next Capability Pointer field. This indicates to software that there is another capability four DWORDs after this capability (for example, at address N+16 where N is the address of this DWORD). If enabled, an external address decoder that snoops the xHC slave interface must be implemented. If it sees an access to N+16 or greater, the slave access is rerouted to a piece of hardware which returns the external capability pointer register of the new capability and also handles reads/writes to this new capability and the side effects. If disabled, a read to the first DWORD of the last internal extended capability returns 0 in the 'Next Capability Pointer' field. This indicates there are no more capabilities.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | <p>InsrtExtrFSBODI Insert Extra Delay Between FS Bulk OUT Some FS devices are slow to receive Bulk OUT data and can get stuck when there are consecutive Bulk OUT transactions with short inter-transaction delays. This bit is used to control whether the host inserts extra delay between consecutive Bulk OUT transactions to a FS Endpoint.</p> <p>1'b0: Host doesn't insert extra delay between consecutive Bulk OUT transactions to a FS Endpoint.</p> <p>1'b1: Host inserts about 12us extra delay between consecutive Bulk OUT transactions to a FS Endpoint to work around the device issue.</p> <p>Note: Setting this bit to one will reduce the Bulk OUT transfer performance for most of the FS devices.</p> |
| 10:9 | RW | 0x0 | <p>DTCT Device Timeout Coarse Tuning This field is a Host mode parameter which determines how long the host waits for a response from device before considering a timeout.</p> <p>The core first checks the DTCT value. If it is 0, then the timeout value is defined by the DTFT. If it is non-zero, then it uses the following timeout values:</p> <p>2'b00: 0 usec -> use DTFT value instead 2'b01: 500 usec 2'b10: 1.5 msec 2'b11: 6.5 msec</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 8:0 | RW | 0x010 | <p>DTFT Device Timeout Fine Tuning This field is a Host mode parameter which determines how long the host waits for a response from device before considering a timeout.</p> <p>For the DTFT field to take effect, DTCT must be set to 2'b00. The DTFT value is the number of 125 MHz clocks * 256 to count before considering a device timeout.</p> <p>The minimum value of DTFT is 2.</p> <p>For example, if the mac3_clk is 125 MHz clk (8 ns period), this is calculated as follows:</p> $(DTFT \text{ value}) * 256 * (8 \text{ ns})$ <p>Quick Reference:</p> <ul style="list-style-type: none"> if DTFT = 0x2, $2*256*8 = 4\text{usec}$ timeout if DTFT = 0x5, $5*256*8 = 10\text{usec}$ timeout if DTFT = 0xA, $10*256*8 = 20\text{usec}$ timeout if DTFT = 0x10, $16*256*8 = 32\text{usec}$ timeout if DTFT = 0x19, $25*256*8 = 51\text{usec}$ timeout if DTFT = 0x31, $49*256*8 = 100\text{usec}$ timeout if DTFT = 0x62, $98*256*8 = 200\text{usec}$ timeout |

USB3_GBUSERRADDRLO

Address: Operational Base + offset (0xc130)

Global SoC Bus Error Address Register - Low

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | <p>BUSERRADDR Bus Address - Low</p> <p>This register contains the lower 32 bits of the first bus address that encountered a SoC bus error. It is valid when the GSTS.BusErrAddrVld field is 1. It can only be cleared by resetting the core.</p> <p>Note: Only supported in AHB and AXI configurations.</p> |

USB3_GBUSERRADDRHI

Address: Operational Base + offset (0xc134)

Global SoC Bus Error Address Register - High

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RU | 0x00000000 | <p>BUSERRADDR Bus Address - High</p> <p>This register contains the higher 32 bits of the first bus address that encountered a SoC bus error. It is valid when the GSTS.BusErrAddrVld field is 1. It can only be cleared by resetting the core.</p> <p>Note: Only supported in AHB and AXI configurations.</p> |

USB3_GPRTBIMAPLO

Address: Operational Base + offset (0xc138)

Global SS Port to Bus Instance Mapping Register - Low

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | BINUM1 SS USB Instance Number for Port 1 Application-programmable ID field. |

USB3_GHWPARAMS0

Address: Operational Base + offset (0xc140)

Global Hardware Parameters Register 0

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x2020400a | GHWPARAMS0 Global Hardware Parameters Register 0 Global Hardware Parameters Register 0 |

USB3_GHWPARAMS1

Address: Operational Base + offset (0xc144)

Global Hardware Parameters Register 1

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x0160c93b | GHWPARAMS1 Global Hardware Parameters Register 1 Global Hardware Parameters Register 1 |

USB3_GHWPARAMS2

Address: Operational Base + offset (0xc148)

Global Hardware Parameters Register 2

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x12345678 | GHWPARAMS2 Global Hardware Parameters Register 2 Global Hardware Parameters Register 2 |

USB3_GHWPARAMS3

Address: Operational Base + offset (0xc14c)

Global Hardware Parameters Register 3

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x069cd085 | GHWPARAMS3 Global Hardware Parameters Register 3 Global Hardware Parameters Register 3 |

USB3_GHWPARAMS4

Address: Operational Base + offset (0xc150)

Global Hardware Parameters Register 4

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x47822008 | GHWPARAMS4 Global Hardware Parameters Register 4 Global Hardware Parameters Register 4 |

USB3_GHWPARAMS5

Address: Operational Base + offset (0xc154)

Global Hardware Parameters Register 5

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x04202088 | GHWPARAMS5 Global Hardware Parameters Register 5 Global Hardware Parameters Register 5 |

USB3_GHWPARAMS6

Address: Operational Base + offset (0xc158)

Global Hardware Parameters Register 6

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x077c8020 | GHWPARAMS6 Global Hardware Parameters Register 6 Global Hardware Parameters Register 6 |

USB3_GHWPARAMS7

Address: Operational Base + offset (0xc15c)

Global Hardware Parameters Register 7

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x03080756 | GHWPARAMS7 Global Hardware Parameters Register 7 Global Hardware Parameters Register 7 |

USB3_GDBGFIFOSPACE

Address: Operational Base + offset (0xc160)

Global Debug Queue/FIFO Space Available Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0042 | SPACE_AVAILABLE Space Available Space Available |
| 15:9 | RO | 0x0 | reserved |
| 8:0 | RW | 0x000 | FIFO_QUEUE_SELECT FIFO/Queue Select (or) Port-Select FIFO/Queue Select[8:5] indicates the FIFO/Queue Type FIFO/Queue Select[4:0] indicates the FIFO/Queue Number Port-Select[3:0] selects the port-number when accessing GDBGLTSSM register. |

USB3_GDBGLTSSM

Address: Operational Base + offset (0xc164)

Global Debug LTSSM Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30 | RO | 0x1 | RxElecidle RxElecidle Reflect status of Pipe interface. |
| 29:27 | RO | 0x0 | reserved |
| 26 | RW | 0x0 | LTDBTIMEOUT LTDB Timeout LTDB Timeout |
| 25:22 | RO | 0x4 | LTDBLINKSTATE LTDB Link State LTDB Link State |
| 21:18 | RO | 0x0 | LTDBSUBSTATE LTDB Sub-State LTDB Sub-State |
| 17 | RO | 0x0 | ELASTICBUFFERMODE ELASTICBUFFERMODE Reflect status of Pipe interface. |
| 16 | RO | 0x1 | TXELECLDLE TXELECLDLE Reflect status of Pipe interface. |
| 15 | RO | 0x0 | RXPOLARITY RXPOLARITY Reflect status of Pipe interface. |
| 14 | RO | 0x0 | TxDetRxLoopback Tx Detect Rx/Loopback Reflect status of Pipe interface. |
| 13:11 | RO | 0x0 | LTDBPhyCmdState LTSSM PHY command State 000: PHY_IDLE (PHY command state is in IDLE. No PHY request pending) 001: PHY_DET (Request to start Receiver detection) 010: PHY_DET_3 (Wait for Phy_Status (Receiver detection)) 011: PHY_PWR_DLY (Delay Pipe3_PowerDown P0 -> P1/P2/P3 request) 100: PHY_PWR_A (Delay for internal logic) 101: PHY_PWR_B (Wait for Phy_Status(Power state change request)) |
| 10:9 | RO | 0x2 | POWERDOWN POWERDOWN Reflect status of Pipe interface. |
| 8 | RO | 0x0 | RXEQTRAIN RXEQTRAIN Reflect status of Pipe interface. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 7:6 | RO | 0x1 | TXDEEMPHASIS TXDEEMPHASIS Reflect status of Pipe interface. |
| 5:3 | RO | 0x0 | LTDBClkState LTSSM Clock State In multi-port host configuration, the port number is defined by Port-Select[3:0] field in the GDBGIFOSPACE register. Note:GDBGLTSSM register is not applicable for USB 2.0-only mode. 000: CLK_NORM (PHY is in non-P3 state and PCLK is running) 001: CLK_TO_P3 (P3 entry request to PHY); 010: CLK_WAIT1 (Wait for Phy_Status (P3 request)); 011: CLK_P3 (PHY is in P3 and PCLK is not running); 100: CLK_TO_P0 (P3 exit request to PHY); 101: CLK_WAIT2 (Wait for Phy_Status (P3 exit request)) |
| 2 | RO | 0x0 | TXSWING TXSWING Reflect status of Pipe interface. |
| 1 | RO | 0x0 | RXTERMINATION RXTERMINATION Reflect status of Pipe interface. |
| 0 | RO | 0x0 | TXONESZEROS TXONESZEROS Reflect status of Pipe interface. |

USB3_GDBGLNMCC

Address: Operational Base + offset (0xc168)

Global Debug LNMCC Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:9 | RO | 0x0 | reserved |
| 8:0 | RO | 0x000 | LNMCC_BERC LNMCC_BERC This field indicates the bit error rate information for the port selected in the GDBGIFOSPACE.PortSelect field.This field is for debug purposes only. |

USB3_GDBGBMU

Address: Operational Base + offset (0xc16c)

Global Debug BMU Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RW | 0x000000 | BMU_BCU BMU_BCU Debug information BMU_BCU Debug information |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 7:4 | RO | 0x0 | BMU_DCU BMU_DCU Debug information BMU_DCU Debug information |
| 3:0 | RO | 0x0 | BMU_CCU BMU_CCU Debug information BMU_CCU Debug information |

USB3_GDBGLSPMUX

Address: Operational Base + offset (0xc170)

Global Debug LSP MUX Register - Device

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x3f | logic_analyzer_trace Logic Analyzer Trace Port MUX Select Currently only bits[21:16] are used. A value of 6'h3F drives "0"s on the logic_analyzer_trace signal. If you plan to OR (instead using a mux) this signal with other trace signals in your system to generate a common trace signal, you can use this feature. |
| 15 | RW | 0x0 | EnDbc EnDbc Enable debugging of Debug capability LSP in Host mode. Use HostSelect to select DbC LSP debug information presented in the GDBGLSP register. |
| 14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | HOSTSELECT Host LSP Select Selects the LSP debug information presented in the GDBGLSP register in host mode. |
| 7:4 | RW | 0x0 | DEVSELECT Device LSP Select Selects the LSP debug information presented in the GDBGLSP register in device mode. Or bit[7:4] of HOSTSELECT, Selects the LSP debug information presented in the GDBGLSP register in host mode. |
| 3:0 | RW | 0x0 | EPSELECT Device Endpoint Select Selects the Endpoint debug information presented in the GDBGEPINFO registers in device mode. Or bit[3:0] of HOSTSELECT, Selects the LSP debug information presented in the GDBGLSP register in host mode. |

USB3_GDBGLSP

Address: Operational Base + offset (0xc174)

Global Debug LSP Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | LSPDEBUG LSP Debug Information LSP Debug Information |

USB3_GDBGEPINFO0

Address: Operational Base + offset (0xc178)
Global Debug Endpoint Information Register 0

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | EPDEBUG Endpoint Debug Information Low 32-bit Endpoint Debug Information Low 32-bit |

USB3_GDBGEPINFO1

Address: Operational Base + offset (0xc17c)
Global Debug Endpoint Information Register 1

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00800000 | EPDEBUG Endpoint Debug Information High 32-bit Endpoint Debug Information High 32-bit |

USB3_GPRTBIMAP_HSLO

Address: Operational Base + offset (0xc180)
Global High-Speed Port to Bus Instance Mapping Register - Low

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | BINUM1 HS USB Instance Number for Port 1 Application-programmable ID field. |

USB3_GPRTBIMAP_FSLO

Address: Operational Base + offset (0xc188)
Global Full-Speed Port to Bus Instance Mapping Register - Low

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | BINUM1 FS USB Instance Number for Port 1 Application-programmable ID field. |

USB3_GUSB2PHYCFG0

Address: Operational Base + offset (0xc200)
Global USB2 PHY Configuration Register 0

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31 | RW | 0x0 | <p>PHYSOFRST UTMI PHY Soft Reset Causes the usb2phy_reset signal to be asserted to reset a UTMI PHY. Not applicable to ULPI because ULPI PHYs are reset via their FunctionControl.Reset register, and the core automatically writes to this register when the core is reset (vcc_reset_n, USBCMD.HCRST, DCTL.SoftReset, or GCTL.SoftReset)</p> |
| 30 | RW | 0x1 | <p>U2_FREECLK_EXISTS U2_FREECLK_EXISTS Specifies whether your USB 2.0 PHY provides a free-running PHY clock, which is active when the clock control input is active. If your USB 2.0 PHY provides a free-running PHY clock, it must be connected to the utmi_clk[0] input. The remaining utmi_clk[n] must be connected to the respective port clocks. The core uses the Port-0 clock for generating the internal mac2 clock. 1'b0: USB 2.0 free clock does not exist 1'b1: USB 2.0 free clock exists Note: When the core is configured as device-only, do not set this bit to 1.</p> |
| 29:25 | RO | 0x0 | reserved |
| 24:22 | RW | 0x0 | <p>LSTRD LS Turnaround Time This field indicates the value of the Rx-to-Tx packet gap for LS devices. The encoding is as follows: 0: 2 bit times 1: 2.5 bit times 2: 3 bit times 3: 3.5 bit times 4: 4 bit times 5: 4.5 bit times 6: 5 bit times 7: 5.5 bit times Note: This field is applicable only in Host mode. For normal operation (to work with most LS devices), set the default value of this field to 3'h0 (2 bit times). The programmable LS device inter-packet gap and turnaround delays are provided to support some legacy LS devices that might require different delays than the default/fixed ones. For instance, the Open LS mouse requires 3 bit times of inter-packet gap to work correctly. Include your PHY delays when programming the LSIPD/LSTRDTIM values. For example, if your PHY's TxEndDelay in LS mode is 30 UTMI/ULPI CLKs, then subtract this delay (~1 LS bit time) from the device's delay requirement.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 21:19 | RW | 0x2 | <p>LSIPD LS Inter-Packet Time This field indicates the value of Tx-to-Tx packet gap for LS devices. The encoding is as follows:</p> <ul style="list-style-type: none"> 0: 2 bit times 1: 2.5 bit times 2: 3 bit times 3: 3.5 bit times 4: 4 bit times 5: 4.5 bit times 6: 5 bit times 7: 5.5 bit times <p>Note: This field is applicable only in Host mode. For normal operation (to work with most LS devices), set the default value of this field to 3'h2 (3 bit times). The programmable LS device inter-packet gap and turnaround delays are provided to support some legacy LS devices that might require different delays than the default/fixed ones. For instance, the AOpen LS mouse requires 3 bit times of inter-packet gap to work correctly. Include your PHY delays when programming the LSIPD/LSTRDTIM values. For example, if your PHY's TxEndDelay in LS mode is 30 UTMI/ULPI CLKs, then subtract this delay (~1 LS bit time) from the device's delay requirement.</p> |
| 18:14 | RO | 0x0 | reserved |
| 13:10 | RW | 0x9 | <p>USBTRDTIM USB 2.0 Turnaround Time Sets the turnaround time in PHY clocks. Specifies the response time for a MAC request to the Packet FIFO Controller (PFC) to fetch data from the DFIFO (SPRAM). The following are the required values for the minimum SoC bus frequency of 60 MHz. USB turnaround time is a critical certification criteria when using long cables and five hub levels. The required values for this field: 4'h5: When the MAC interface is 16-bit UTMI+. 4'h9: When the MAC interface is 8-bit UTMI+/ULPI. If SoC bus clock is less than 60 MHz, and USB turnaround time is not critical, this field can be set to a larger value. Note: This field is valid only in device mode.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 9 | RW | 0x0 | <p>XCVRDLY Transceiver Delay Enables a delay between the assertion of the UTMI/ULPI Transceiver Select signal (for HS) and the assertion of the TxValid signal during a HS Chirp.</p> <p>When this bit is set to 1, a delay (of approximately 2.5 us) is introduced from the time when the Transceiver Select is set to 2'b00 (HS) to the time the TxValid is driven to 0 for sending the chirp-K.</p> <p>This delay is required for some UTMI/ULPI PHYs.</p> <p>Note:</p> <p>If you enable the hibernation feature when the device core comes out of power-off, you must re-initialize this bit with the appropriate value because the core does not save and restore this bit value during hibernation.</p> <p>This bit is valid only in device mode.</p> |
| 8 | RW | 0x0 | <p>ENBLSLPM Enable utmi_sleep_n and utmi_l1_suspend_n The application uses this bit to control utmi_sleep_n and utmi_l1_suspend_n assertion to the PHY in the L1 state.</p> <p>1'b0: utmi_sleep_n and utmi_l1_suspend_n assertion from the core is not transferred to the external PHY.</p> <p>1'b1: utmi_sleep_n and utmi_l1_suspend_n assertion from the core is transferred to the external PHY.</p> <p>Note:</p> <p>This bit must be set high for Port0 if SNPS PHY is used.</p> <p>In Device mode - Before issuing any device endpoint command when operating in 2.0 speeds, disable this bit and enable it after the command completes. Without disabling this bit, if a command is issued when the device is in L1 state and if mac2_clk (utmi_clk/ulpi_clk) is gated off, the command will not get completed.</p> |
| 7 | RO | 0x0 | <p>PHYSEL USB 2.0 High-Speed PHY or USB 1.1 Full-Speed</p> <p>1'b0: USB 2.0 high-speed UTMI+ or ULPI PHY.</p> <p>1'b1: USB 1.1 full-speed serial transceiver.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 6 | RW | 0x0 | <p>SUSPENDUSB20 Suspend USB2.0 HS/FS/LS PHY When set, USB2.0 PHY enters Suspend mode if Suspend conditions are valid. For DRD/OTG configurations, it is recommended that this bit is set to 0 during coreConsultant configuration. If it is set to 1, then the application must clear this bit after power-on reset. Application needs to set it to 1 after the core initialization completes.</p> <p>For all other configurations, this bit can be set to 1 during core configuration.</p> <p>Note: In host mode, on reset, this bit is set to 1. Software can override this bit after reset. In device mode, before issuing any device endpoint command when operating in 2.0 speeds, disable this bit and enable it after the command completes. If you issue a command without disabling this bit when the device is in L2 state and if mac2_clk (utmi_clk/ulpi_clk) is gated off, the command will not get completed.</p> |
| 5 | RO | 0x0 | reserved |
| 4 | RO | 0x0 | <p>ULPI_UTMI_Sel ULPI or UTMI+ Select 1'b0: UTMI+ Interface 1'b1: ULPI Interface</p> |
| 3 | RW | 0x0 | <p>PHYIF PHY Interface If UTMI+ is selected, the application uses this bit to configure the core to support a UTMI+ PHY with an 8- or 16-bit interface. 1'b0: 8 bits 1'b1: 16 bits</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 2:0 | RW | 0x0 | <p>TOutCal HS/FS Timeout Calibration</p> <p>The number of PHY clocks, as indicated by the application in this field, is multiplied by a bit-time factor; this factor is added to the high-speed/full-speed interpacket timeout duration in the core to account for additional delays introduced by the PHY. This may be required, since the delay introduced by the PHY in generating the linestate condition may vary among PHYs.</p> <p>The USB standard timeout value for high-speed operation is 736 to 816 (inclusive) bit times. The USB standard timeout value for full-speed operation is 16 to 18 (inclusive) bit times. The application must program this field based on the speed of connection. The number of bit times added per PHY clock are:</p> <p>High-speed operation:</p> <ul style="list-style-type: none"> One 30-MHz PHY clock = 16 bit times One 60-MHz PHY clock = 8 bit times <p>Full-speed operation:</p> <ul style="list-style-type: none"> One 30-MHz PHY clock = 0.4 bit times One 60-MHz PHY clock = 0.2 bit times |

USB3_GUSB3PIPECTL0

Address: Operational Base + offset (0xc2c0)

Global USB3 PIPE Control Register 0

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 31 | RW | 0x0 | <p>PHYSoftRst USB3 PHY Soft Reset</p> <p>After setting this bit to 1, the software needs to clear this bit.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 30 | RW | 0x0 | <p>HstPrtCmpl HstPrtCmpl</p> <p>This feature tests the PIPE PHY compliance patterns without having to have a test fixture on the USB 3.0 cable.</p> <p>This bit enables placing the SS port link into a compliance state. By default, this bit must be set to 1'b0.</p> <p>In compliance lab testing, the SS port link enters compliance after failing the first polling sequence after power on. Set this bit to 0, when you run compliance tests.</p> <p>The sequence for using this functionality is as follows:</p> <ol style="list-style-type: none"> 1. Disconnect any plugged in devices. 2. Perform USBCMD.HCRST or power-on-chip reset. 3. Set PORTSC.PP=0. 4. Set GUSB3PIPECTL. HstPrtCmpl=1. This places the link into compliance state. <p>To advance the compliance pattern, follow this sequence (toggle the set GUSB3PIPECTL. HstPrtCmpl):</p> <ol style="list-style-type: none"> 1. Set GUSB3PIPECTL.HstPrtCmpl=0. 2. Set GUSB3PIPECTL.HstPrtCmpl=1. This advances the link to the next compliance pattern. <p>To exit from the compliance state perform USBCMD.HCRST or power-on-chip reset.</p> |
| 29 | RW | 0x0 | <p>U2SSInactP3ok P3 OK for U2/SSInactive</p> <p>0: During link state U2/SS.Inactive, put PHY in P2 (Default)</p> <p>1: During link state U2/SS.Inactive, put PHY in P3.</p> <p>Note: For a port, if GUSB3PIPECTL[7]=1 and GUSB3PIPECTL[29]=1, set GUSB3PIPECTL[11] to 1.</p> |
| 28 | RW | 0x0 | <p>DisRxDetP3 Disabled receiver detection in P3</p> <p>0: If PHY is in P3 and Core needs to perform receiver detection, The core performs receiver detection in P3. (Default)</p> <p>1: If PHY is in P3 and Core needs to perform receiver detection, The core changes the PHY power state to P2 and then performs receiver detection. After receiver detection, the cores changes PHY power state to P3.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 27 | RW | 0x0 | <p>Ux_exit_in_Px Ux Exit in Px 0: The core does U1/U2/U3 exit in PHY power state P0 (default behavior). 1: The core does U1/U2/U3 exit in PHY power state P1/P2/P3 respectively. This bit is added for SS PHY workaround where SS PHY injects a glitch on pipe3_RxElecIdle while receiving Ux exit LFPS, and pipe3_PowerDown change is in progress. Note: This bit is used by third-party SS PHY. It must be set to 0 for SNPS PHY.</p> |
| 26 | RW | 0x0 | <p>ping_enhancement_en Ping Enhancement Enable When set, the Downstream port U1 ping receive timeout becomes 500 ms instead of 300 ms. Minimum Ping.LFPS receive duration is 8 ns (one mac3_clk). This field is valid for the downstream port only. Note: This bit is used by third-party SS PHY. It must be set to 0 for SNPS PHY.</p> |
| 25 | RW | 0x0 | <p>u1u2exitfail_to_recov U1U2exitfail to Recovery When set, and U1/U2 LFPS handshake fails, the LTSSM transitions from U1/U2 to Recovery instead of SS.Inactive. If Recovery fails, then the LTSSM can enter SS.Inactive. This is an enhancement only. It prevents interoperability issue if the remote link does not do proper handshake.</p> |
| 24 | RW | 0x1 | <p>request_p1p2p3 Always Request P1/P2/P3 for U1/U2/U3 When set, the core always requests PHY power change from P0 to P1/P2/P3 during U0 to U1/U2/U3 transition. If this bit is 0, and immediate Ux exit (remotely initiated, or locally initiated) happens, the core does not request P1/P2/P3 power state change. Note: This bit must be set to 1 for SNPS PHY. For third-party SS PHY, check with your PHY vendor.</p> |
| 23 | RW | 0x0 | <p>StartRxDetU3RxDet Start Receiver Detection in U3/Rx.Detect If DWC_USB3_GUSB3PIPECTL_INIT[22] is set, and the link is in either U3 or Rx.Detect state, the core starts receiver detection on the rising edge of this bit. This can only be used for Downstream ports. This bit must be set to 0 for Upstream ports. This feature must not be enabled for normal operation. If have to use this feature, contact SNPS.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 22 | RW | 0x0 | <p>DisRxDetU3RxDet Disable Receiver Detection in U3/Rx.Det When set, the core does not handle receiver detection in either U3 or Rx.Detect states. DWC_USB3_GUSB3PIPECTL_INIT[23] must be used to start receiver detection manually. This bit can only be used for the downstream port. This bit must be set to 0 for Upstream ports. This feature must not be enabled for normal operation. If you have to use this feature, contact SNPS.</p> |
| 21:19 | RW | 0x1 | <p>DelayP1P2P3 Delay P1P2P3 Delay P0 to P1/P2/P3 request when entering U1/U2/U3 until (DWC_USB3_GUSB3PIPECTL_INIT[21:19]*8) 8B10B error occurs, or Pipe3_RxValid drops to 0. DWC_USB3_GUSB3PIPECTL_INIT[18] must be 1 to enable this functionality.</p> |
| 18 | RW | 0x1 | <p>DELAYP1TRANS DELAYP1TRANS Delay PHY power change from P0 to P1/P2/P3 when link state changing from U0 to U1/U2/U3 respectively. 1'b1: When entering U1/U2/U3, delay the transition to P1/P2/P3 until the pipe3 signals, Pipe3_RxElecIlde is 1 and pipe3_RxValid is 0 1'b0: When entering U1/U2/U3, transition to P1/P2/P3 without checking for Pipe3_RxElecIlde and pipe3_RxValid. Note: This bit must be set to '1' for SNPS PHY. It is also used by third-party SS PHY.</p> |
| 17 | RW | 0x0 | <p>SUSPENDENABLE Suspend USB3.0 SS PHY When set, and if Suspend conditions are valid, the USB 3.0 PHY enters Suspend mode. For DRD/OTG configurations, it is recommended that this bit is set to '0' during coreConsultant configuration. If it is set to '1', then the application must clear this bit after power-on reset. Application needs to set it to '1' after the core initialization is completed. For all other configurations, this bit can be set to '1' during core configuration.</p> |
| 16:15 | RW | 0x0 | <p>DATWIDTH PIPE Data Width 2'b00: 32 bits 2'b01: 16 bits 2'b10: 8 bits Note: USB3 controller only support 32-bit width pipe interface.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 14 | RW | 0x0 | <p>AbortRxDetInU2 Abort Rx Detect in U2 When set and the link state is U2, then the core will abort receiver detection if it receives U2 exit LFPS from the remote link partner. This bit is for the downstream port only.</p> <p>Note: This bit is used by third-party SS PHY. It must be set to '0' for SNPS PHY.</p> |
| 13 | RW | 0x0 | <p>SkipRxDet Skip Rx Detect When set, the core skips Rx Detection if pipe3_RxElecIdle is low. Skip is defined as waiting for the appropriate timeout, then repeating the operation.</p> |
| 12 | RW | 0x0 | <p>LFPSP0Align LFPS P0 Align When set: 1. The core deasserts LFPS transmission on the clock edge that it requests Phy power state 0 when exiting U1, U2, or U3 low power states. Otherwise, LFPS transmission is asserted one clock earlier. 2. The core requests symbol transmission two pipe3_rx_pclks periods after the PHY asserts PhyStatus as a result of the PHY switching from P1 or P2 state to P0 state. Currently, this bit is only used in USB 3.0 HUB with SNPS PHY. For other USB 3.0 Host, Device, and DRD cores, this bit is not required.</p> |
| 11 | RW | 0x0 | <p>P3P2TranOK P3 P2 Transitions OK When set, the core transitions directly from Phy power state P2 to P3 or from state P3 to P2. When not set, P0 is always entered as an intermediate state during transitions between P2 and P3, as defined in the PIPE3 Specification. According to the PIPE3 Specification, any direct transition between P3 and P2 is illegal. Note: This bit is used by third-party SS PHY. It must be set to '0' for SNPS PHY.</p> |
| 10 | RW | 0x0 | <p>P3ExSigP2 P3 Exit Signal in P2 When this bit is set, the core always changes the PHY power state to P2, before attempting a U3 exit handshake. This bit is used only for some non-SNPS PHYs that cannot do LFPS in P3. Note: This bit is used by third-party SS PHY. It must be set to '0' for SNPS PHY.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9 | RW | 0x0 | LFPSFILTER LFPS Filter When set, filter LFPS reception with pipe3_RxValid in PHY power state P0, that is, ignore LFPS reception from the PHY unless both pipe3_Rxelecidle and pipe3_RxValid are deasserted. |
| 8 | RW | 0x0 | RX_DETECT_to_Polling_L RX_DETECT to Polling.LFPS Control 1'b0 (Default): Enables a 400us delay to start Polling LFPS after RX_DETECT. This allows VCM offset to settle to a proper level. 1'b1: Disables the 400us delay to start Polling LFPS after RX_DETECT. During controller certification with third party PHY it is observed that the PHY is not able to meet the Tx AC common mode voltage active (VTX-CM-ACPP_ACTIVE <100mv) if the link starts polling within 80us from the time rx.detect is performed. To meet this VTX-CM-ACPP_ACTIVE specification, the polling must be delayed further. If the PHY does not have issue then they can set this bit to 1 which allows polling to start within 80us. |
| 7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | TX_SWING Tx Swing Drive the setting value to the pipe interface of PHY. |
| 5:3 | RW | 0x0 | TX_MARGIN Tx Margin[2:0] Drive the setting value to the pipe interface of PHY. |
| 2:1 | RW | 0x1 | TX_DE_EPPHASIS Tx Deemphasis The value driven to the PHY is controlled by the LTSSM during USB3 Compliance mode. |
| 0 | RW | 0x0 | ELASTIC_BUFFER_MODE Elastic Buffer Mode Drive the setting value to the pipe interface of PHY. |

USB3_GTXFIFOSIZn

Address: Operational Base + offset (0xc300 + 4*n), n=0~6

Global Transmit FIFO Size Register n

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | TXFSTADDR_N Transmit FIFO RAM Start Address This field contains the memory start address for TxFIFO in 64-bit words. |
| 15:0 | RW | 0x0042 | TXFDEP_N TxFIFO Depth This field contains the depth of TxFIFO in 64-bit words. Minimum value: 32; Maximum value: 32,768 |

USB3_GRXFIFOSIZn

Address: Operational Base + offset (0xc380 + 4*n), n=0~2

Global Receive FIFO Size Register n

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | RXFSTADDR_N RxFIFOOn RAM Start Address This field contains the memory start address for RxFIFOOn in 64-bit words. |
| 15:0 | RW | 0x0285 | RXFDEP_N RxFIFO Depth This field contains the depth of RxFIFOOn in 64-bit words. Minimum value: 32; Maximum value: 16,384 |

USB3_GEVNTADRLO0

Address: Operational Base + offset (0xc400)

Global Event Buffer Address (Low) Register 0

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EVNTADRLO Event Buffer Address Holds the lower 32 bits of start address of the external memory for the Event Buffer. During operation, hardware does not update this address. |

USB3_GEVNTADRHI0

Address: Operational Base + offset (0xc404)

Global Event Buffer Address (High) Register 0

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | EVNTADRHI Event Buffer Address Holds the higher 32 bits of start address of the external memory for the Event Buffer. During operation, hardware does not update this address. |

USB3_GEVNTSIZ0

Address: Operational Base + offset (0xc408)

Global Event Buffer Size Register 0

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | EVNTINTRPTMASK Event Interrupt Mask When set to '1', this prevents the interrupt from being generated. However, even when the mask is set, the events are queued. |
| 30:16 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 15:0 | RW | 0x0000 | EVENTSIZ Event Buffer Size in bytes Holds the size of the Event Buffer in bytes; must be a multiple of four. This is programmed by software once during initialization. The minimum size of the event buffer is 32 bytes. |

USB3_GEVNTCOUNT0

Address: Operational Base + offset (0xc40c)

Global Event Buffer Count Register 0

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15:0 | RW | 0x0000 | EVNTCOUNT Event Count When read, returns the number of valid events in the Event Buffer (in bytes). When written, hardware decrements the count by the value written. The interrupt line remains high when count is not 0. |

USB3_GHWPARAMS8

Address: Operational Base + offset (0xc600)

Global Hardware Parameters Register 8

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000077c | ghwparams8_32_0 ghwparams8_32_0 ghwparams8_32_0 |

USB3_GTXFIFOPRIDEV

Address: Operational Base + offset (0xc610)

Global Device TX FIFO DMA Priority Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:7 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 6:0 | RW | 0x00 | <p>gtxfifopridev Device TxFIFO priority</p> <p>This register specifies the relative DMA priority level among the Device TXFIFOs (one per IN endpoint). Each register bit[n] controls the priority (1: high, 0: low) of each TXFIFO[n]. When multiple TXFIFOs compete for DMA service at a given time (that is, multiple TXQs contain TX DMA requests and their corresponding TXFIFOs have space available), the TX DMA arbiter grants access on a packet-basis in the following manner:</p> <ol style="list-style-type: none"> 1. High-priority TXFIFOs are granted access using round-robin arbitration 2. Low-priority TXFIFOs are granted access using round-robin arbitration only after the high-priority TXFIFOs have no further processing to do (that is, either the TXQs are empty or the corresponding TXFIFOs are full). <p>For scatter-gather packets, the arbiter grants successive DMA requests to the same FIFO until the entire packet is completed. When configuring periodic IN endpoints, software must set register bit[n]=1, where n is the TXFIFO assignment. This ensures that the DMA for isochronous or interrupt IN endpoints are prioritized over bulk or control IN endpoints.</p> <p>This register is present only when the core is configured to operate in the device mode. The register size corresponds to the number of Device IN endpoints.</p> |

USB3_GTXFIFOPRIHST

Address: Operational Base + offset (0xc618)

Global Host TX FIFO DMA Priority Register

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:3 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 2:0 | RW | 0x0 | <p>gtxfifoprihst Host TxFIFO priority</p> <p>This register specifies the relative DMA priority level among the Host TXFIFOs (one per USB bus instance) within the associated speed group (SS or HS/FSLS). Each register bit[n] controls the priority (1: high, 0: low) of TXFIFO[n] within a speed group.</p> <p>When multiple TXFIFOs compete for DMA service at a given time (i.e., multiple TXQs contain TX DMA requests and their corresponding TXFIFOs have space available), the TX DMA arbiter grants access on a packet-basis in the following manner:</p> <ol style="list-style-type: none"> 1. Among the FIFOs in the same speed group (SS or HS/FSLS): <ol style="list-style-type: none"> a. High-priority TXFIFOs are granted access using round-robin arbitration b. Low-priority TXFIFOs are granted access using round-robin arbitration only after the high-priority TXFIFOs have no further processing to do (that is, either the TXQs are empty or the corresponding TXFIFOs are full). 2. The TX DMA arbiter prioritizes the SS speed group or HS/FSLS speed group according to the ratio programmed in the GDMAHLRATIO register. <p>For scatter-gather packets, the arbiter grants successive DMA requests to the same FIFO until the entire packet is completed.</p> <p>This register is present only when the core is configured to operate in the host mode (includes DRD and OTG modes). The register size corresponds to the number of configured USB bus instances; for example, in the default configuration, there are 3 USB bus instances (1 SS, 1 HS, and 1 FSLS).</p> |

USB3_GRXFIFOPRIHST

Address: Operational Base + offset (0xc61c)

Global Host RX FIFO DMA Priority Register

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:3 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 2:0 | RW | 0x0 | <p>grxfifoprihst Host RxFIFO priority</p> <p>This register specifies the relative DMA priority level among the Host RXFIFOs (one per USB bus instance) within the associated speed group (SS or HS/FSLS). Each register bit[n] controls the priority (1: high, 0: low) of RXFIFO[n] within a speed group.</p> <p>When multiple RXFIFOs compete for DMA service at a given time (i.e., multiple RXQs contain RX DMA requests and their corresponding RXFIFOs have data available), the RX DMA arbiter grants access on a packet-basis in the following manner:</p> <ol style="list-style-type: none"> 1. Among the FIFOs in the same speed group (SS or HS/FSLS): <ol style="list-style-type: none"> a. High-priority RXFIFOs are granted access using round-robin arbitration b. Low-priority RXFIFOs are granted access using round-robin arbitration only after high-priority RXFIFOs have no further processing to do (that is, either the RXQs are empty or the corresponding RXFIFOs do not have the required data). 2. The RX DMA arbiter prioritizes the SS speed group or HS/FSLS speed group according to the ratio programmed in the GDMAHLRATIO register. <p>For scatter-gather packets, the arbiter grants successive DMA requests to the same FIFO until the entire packet is completed.</p> <p>This register is present only when the core is configured to operate in the host mode (includes DRD and OTG modes). The register size corresponds to the number of configured USB bus instances; for example, in the default configuration, there are 3 USB bus instances (1 SS, 1 HS, and 1 FSLS).</p> |

USB3_GFIFOPRIDBC

Address: Operational Base + offset (0xc620)

Global Host Debug Capability DMA Priority Register

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:2 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 1:0 | RW | 0x0 | <p>gfifopridbc Host DbC DMA priority This register specifies the relative priority of the RXFIFOs and TXFIFOs associated with the DbC mode. It overrides the priority assigned in the corresponding indexes of the Host RXFIFO and TXFIFO DMA priority registers, when the DbC mode is enabled. Priority settings are specified in relation to the low-priority SS speed group:</p> <ol style="list-style-type: none"> 1. Normal priority indicates that the DbC FIFOs are considered identical to the Host SS low-priority FIFOs. 2. Low priority indicates that the DbC FIFOs are considered to have lower priority than all Host SS FIFOs. 3. High priority indicates that the DbC FIFOs are considered higher priority than the Host SS low-priority FIFOs but lower priority than the Host SS high-priority FIFOs. <p>This register is present only when the core is configured to operate in Host Debug Capability (DbC) mode.</p> |

USB3_GDMAHLRATIO

Address: Operational Base + offset (0xc624)

Global Host FIFO DMA High-Low Priority Ratio Register

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:13 | RO | 0x0 | reserved |
| 12:8 | RW | 0x00 | <p>hstrxfifo Host RXFIFO DMA High-Low Priority This register specifies the relative priority of the SS FIFOs with respect to the HS/FSLS FIFOs. The DMA arbiter prioritizes the HS/FSLS round-robin arbiter group every DMA High-Low Priority Ratio grants as indicated in the register separately for TX and RX. To illustrate, consider that all FIFOs are requesting access simultaneously, and the ratio is 4. SS gets priority for 4 packets, HS/FSLS gets priority for 1 packet, SS gets priority for 4 packets, HS/FSLS gets priority for 1 packet, and so on. If FIFOs from both speed groups are not requesting access simultaneously then,</p> <ol style="list-style-type: none"> 1. if SS got grants 4 out of the last 4 times, then HS/FSLS get the priority on any future request. 2. if HS/FSLS got the grant last time, SS gets the priority on the next request. 3. if there is a valid request on either SS or HS/FSLS, a grant is always awarded; there is no idle. <p>This register is present if the core is configured to operate in host mode.</p> |
| 7:5 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 4:0 | RW | 0x08 | <p>hsttxfifo Host TXFIFO DMA High-Low Priority</p> <p>This register specifies the relative priority of the SS FIFOs with respect to the HS/FSLS FIFOs. The DMA arbiter prioritizes the HS/FSLS round-robin arbiter group every DMA High-Low Priority Ratio grants as indicated in the register separately for TX and RX.</p> <p>To illustrate, consider that all FIFOs are requesting access simultaneously, and the ratio is 4. SS gets priority for 4 packets, HS/FSLS gets priority for 1 packet, SS gets priority for 4 packets, HS/FSLS gets priority for 1 packet, and so on.</p> <p>If FIFOs from both speed groups are not requesting access simultaneously then,</p> <ol style="list-style-type: none"> 1. if SS got grants 4 out of the last 4 times, then HS/FSLS get the priority on any future request. 2. if HS/FSLS got the grant last time, SS gets the priority on the next request. 3. if there is a valid request on either SS or HS/FSLS, a grant is always awarded; there is no idle. <p>This register is present if the core is configured to operate in host mode.</p> |

USB3_GFLADJ

Address: Operational Base + offset (0xc630)

Global Frame Length Adjustment Register

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 31 | RW | 0x0 | <p>GFLADJ_REFCLK_240MHZDECR_PLS1 GFLADJ_REFCLK_240MHZDECR_PLS1</p> <p>This field indicates that the decrement value that the controller applies for each ref_clk must be GFLADJ_REFCLK_240MHZ_DECR and GFLADJ_REFCLK_240MHZ_DECR + 1 alternatively on each ref_clk.</p> <p>Set this bit to a 1 only if GFLADJ_REFCLK_LPM_SEL is set to 1 and the fractional component of 240/ref_frequency is greater than or equal to 0.5.</p> <p>Examples:</p> <p>If the ref_clk is 24 MHz then</p> <ol style="list-style-type: none"> 1. GUCTL.REF_CLK_PERIOD = 41 2. GFLADJ.GFLADJ_REFCLK_240MHZ_DECR = (240/24) = 10 3. GFLADJ.GFLADJ_REFCLK_240MHZDECR_PLS1 = 0 |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 30:24 | RW | 0x00 | <p>GFLADJ_REFCLK_240MHZ_DECR GFLADJ_REFCLK_240MHZ_DECR</p> <p>This field indicates the decrement value that the controller applies for each ref_clk in order to derive a frame timer in terms of a 240-MHz clock.</p> <p>This field must be programmed to a non-zero value only if GFLADJ_REFCLK_LPM_SEL is set to 1.</p> <p>The value is derived as follows:</p> $\text{GFLADJ_REFCLK_240MHZ_DECR} = 240/\text{ref_clk_frequency}$ <p>Examples: If the ref_clk is 24 MHz then</p> <ol style="list-style-type: none"> 1. GUCTL.REF_CLK_PERIOD = 41 2. GFLADJ.GFLADJ_REFCLK_240MHZ_DECR = $240/24 = 10$ |
| 23 | RW | 0x0 | <p>GFLADJ_REFCLK_LPM_SEL GFLADJ_REFCLK_LPM_SEL</p> <p>This bit enables the functionality of running SOF/ITP counters on the ref_clk. This bit must not be set to 1 if GCTL.SOFITPSYNC bit is set to 1. Similarly, if GFLADJ_REFCLK_LPM_SEL set to 1, GCTL.SOFITPSYNC must not be set to 1.</p> <p>When GFLADJ_REFCLK_LPM_SEL is set to 1 the overloading of the suspend control of the USB 2.0 first port PHY (UTMI/ULPI) with USB 3.0 port states is removed.</p> <p>For example, for SNPS PHY, the COMMONONN signal can be tied to 1.</p> <p>Note that the ref_clk frequencies supported in this mode are 16/17/19.2/20/24/39.7/40 MHz. The utmi_clk[0] signal of the core must be connected to the FREECLK of the PHY.</p> <p>Note: If you set this bit to 1, the GUSB2PHYCFG.U2_FREECLK_EXISTS bit must be set to 0.</p> |
| 22 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 21:8 | RW | 0x0000 | <p>GFLADJ_REFCLK_FLADJ GFLADJ_REFCLK_FLADJ</p> <p>This field indicates the frame length adjustment to be applied when SOF/ITP counter is running on the ref_clk.</p> <p>This register value is used to adjust the ITP interval when GCTL[SOFITPSYNC] is set to 1; SOF and ITP interval when GLADJ.GFLADJ_REFCLK_LPM_SEL is set to 1.</p> <p>This field must be programmed to a non-zero value only if GFLADJ_REFCLK_LPM_SEL is set to 1 or GCTL.SOFITPSYNC is set to 1.</p> <p>The value is derived as follows:</p> $\text{FLADJ_REF_CLK_FLADJ} = ((125000/\text{ref_clk_period_integer}) - (125000/\text{ref_clk_period})) * \text{ref_clk_period}$ <p>where:</p> <ol style="list-style-type: none"> 1. The ref_clk_period_integer is the integer value of the ref_clk period got by truncating the decimal (fractional) value that is programmed in the GUCTL.REF_CLK_PERIOD field. 2. The ref_clk_period is the ref_clk period including the fractional value. <p>Examples: If the ref_clk is 24 MHz then</p> <ol style="list-style-type: none"> 1. GUCTL.REF_CLK_PERIOD = 41 2. GFLADJ.GLADJ_REFCLK_FLADJ = $((125000/41) - (125000/41.6666)) * 41.6666 = 2032$ (ignoring the fractional value) |
| 7 | RW | 0x0 | <p>GFLADJ_30MHZ_SDBND_SEL GFLADJ_30MHZ_SDBND_SEL</p> <p>This field selects whether to use the input signal fladj_30mhz_reg or the GFLADJ.GFLADJ_30MHZ to adjust the frame length for the SOF/ITP. When this bit is set to:</p> <ul style="list-style-type: none"> 1, the controller uses the register field GFLADJ.GFLADJ_30MHZ value 0, the controller uses the input signal fladj_30mhz_reg value |
| 6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x00 | <p>GFLADJ_30MHZ GFLADJ_30MHZ</p> <p>This field indicates the value that is used for frame length adjustment instead of considering from the sideband input signal fladj_30mhz_reg.</p> <p>This enables post-silicon frame length adjustment in case the input signal fladj_30mhz_reg is connected to a wrong value or is not valid.</p> <p>For details on how to set this value, refer to section 5.2.4, "Frame Length Adjustment Register (FLADJ)," of the xHCI Specification.</p> |

USB3_DCFG

Address: Operational Base + offset (0xc700)

Device Configuration Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | <p>IgnStrmPP IgnoreStreamPP</p> <p>This bit only affects stream-capable bulk endpoints.</p> <p>When this bit is set to 0 and the controller receives a Data Packet with the Packet Pending (PP) bit set to 0 for OUT endpoints, or it receives an ACK with the NumP field set to 0 and PP set to 0 for IN endpoints, the core attempts to search for another stream (CStream) to initiate to the host. However, there are two situations where this behavior is not optimal:</p> <ol style="list-style-type: none"> 1. When the host is setting PP=0 even though it has not finished the stream, or 2. When the endpoint on the device is configured with one transfer resource and therefore does not have any other streams to initiate to the host. <p>When this bit is set to 1, the core ignores the Packet Pending bit for the purposes of stream selection and does not search for another stream when it receives DP(PP=0) or ACK(NumP=0, PP=0). This can enhance the performance when the device system bus bandwidth is low or the host responds to the core's ERDY transmission very quickly.</p> |
| 22 | RW | 0x0 | <p>LPMCAP LPM Capable</p> <p>The application uses this bit to control the DWC_usb3 core LPM capabilities. If the core operates as a non-LPM-capable device, it cannot respond to LPM transactions.</p> <p>1'b0: LPM capability is not enabled. 1'b1: LPM capability is enabled.</p> |
| 21:17 | RW | 0x04 | <p>NUMP</p> <p>Number of Receive Buffers.</p> <p>This bit indicates the number of receive buffers to be reported in the ACK TP.</p> <p>The DWC_usb3 controller uses this field if GRXTHRCFG.USBRxPktCntSel is set to 0. The application can program this value based on RxFIFO size, buffer sizes programmed in descriptors, and system latency.</p> <p>For an OUT endpoint, this field controls the number of receive buffers reported in the NumP field of the ACK TP transmitted by the core.</p> <p>Note: This bit is used in host mode when Debug Capability is enabled.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 16:12 | RW | 0x00 | INTRNUM Interrupt number Indicates interrupt/EventQ number on which non-endpoint-specific device-related interrupts (see DEVT) are generated. |
| 11:10 | RO | 0x0 | reserved |
| 9:3 | RW | 0x00 | DEVADDR Device Address. The application must perform the following: 1. Program this field after every SetAddress request. 2. Reset this field to zero after USB reset. |
| 2:0 | RW | 0x4 | DEVSPD Device Speed Indicates the speed at which the application requires the core to connect, or the maximum speed the application can support. However, the actual bus speed is determined only after the chirp sequence is completed, and is based on the speed of the USB host to which the core is connected. 3'b100: SuperSpeed (USB 3.0 PHY clock is 125 MHz or 250 MHz) 3'b000: High-speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) 3'b001: Full-speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) |

USB3_DCTL

Address: Operational Base + offset (0xc704)

Device Control Register

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 31 | RW | 0x0 | <p>RUN_STOP Run/Stop</p> <p>The software writes 1 to this bit to start the device controller operation.</p> <p>To stop the device controller operation, the software must remove any active transfers and write 0 to this bit. When the controller is stopped, it sets the DSTS.DevCtrlHlt bit when the core is idle and the lower layer finishes the disconnect process.</p> <p>The Run/Stop bit must be used in following cases as specified:</p> <ol style="list-style-type: none"> 1. After power-on reset and CSR initialization, the software must write 1 to this bit to start the device controller. The controller does not signal connect to the host until this bit is set. 2. The software uses this bit to control the device controller to perform a soft disconnect. When the software writes 0 to this bit, the host does not see that the device is connected. The device controller stays in the disconnected state until the software writes 1 to this bit. <p>The minimum duration of keeping this bit cleared is specified in the Note below. If the software attempts a connect after the soft disconnect or detects a disconnect event, it must set DCTL[8:5] to 5 before reasserting the Run/Stop bit.</p> <ol style="list-style-type: none"> 3. When the USB or Link is in a lower power state and the Two Power Rails configuration is selected, software writes 0 to this bit to indicate that it is going to turn off the Core Power Rail. After the software turns on the Core Power Rail again and re-initializes the device controller, it must set this bit to start the device controller. <p>Note: The following is the minimum duration under various conditions for which the soft disconnect (SftDiscon) bit must be set for the USB host to detect a device disconnect:</p> <p>30ms: For SuperSpeed, when the device state is Suspended, Idle, Transmit, or Receive.</p> <p>10ms: For high-speed, when the device state is Suspended, Idle, or not Idle/Suspended (performing transactions); For full-speed/low-speed, when the device state is Suspended, Idle, or not Idle/Suspended (performing transactions)</p> <p>To accommodate clock jitter, it is recommended that the application add extra delay to the specified minimum duration.</p> |

| Bit | Attr | Reset Value | Description |
|-------|-----------|-------------|--|
| 30 | R/W SC | 0x0 | <p>CSFTRST Core Soft Reset Reset all clock domains as follows:</p> <ol style="list-style-type: none"> 1. This bit clears the interrupts and all the CSRs except GSTS, GSNPSSID, GPIO, GUID, GUSB2PHYCFGn registers, GUSB3PIPECTLn registers, DCFG, DCTL, DEVREN, and DSTS registers. 2. All module state machines (except the SoC Bus Slave Unit) are reset to the IDLE state, and all the TxFIFOs and the RxFIFO are flushed. 3. Any transactions on the SoC bus Master are terminated as soon as possible, after gracefully completing the last data phase of a SoC bus transfer. Any transactions on the USB are terminated immediately. <p>The application can write this bit at any time to reset the core. This is a self-clearing bit; the core clears this bit after all necessary logic is reset in the core, which may take several clocks depending on the core's current state. Once this bit is cleared, the software must wait at least 3 PHY clocks before accessing the PHY domain (synchronization delay).</p> <p>Typically, software reset is used during software development and also when you dynamically change the PHY selection bits in the USB configuration registers listed above. When you change the PHY, the corresponding clock for the PHY is selected and used in the PHY domain. Once a new clock is selected, the PHY domain must be reset for proper operation.</p> |
| 29 | RO | 0x0 | reserved |
| 28:24 | RW | 0x00 | <p>HIRDTHRES HIRD Threshold The core asserts output signals utmi_l1_suspend_n and utmi_sleep_n on the basis of this signal: The core asserts utmi_l1_suspend_n to put the PHY into Deep Low-Power mode in L1 when both of the following are true: 1. HIRD value is greater than or equal to the value in DCTL.HIRD_Thres[3:0] 2. HIRD_Thres[4] is set to 1'b1. The core asserts utmi_sleep_n on L1 when one of the following is true: 1. If the HIRD value is less than HIRD_Thres[3:0] or 2. HIRD_Thres[4] is set to 1'b0. Note: This field must be set to '0' during SuperSpeed mode of operation.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23:20 | RW | 0xf | <p>LPM_NYET_thres LPM NYET Threshold Handshake response to LPM token specified by device application. Response depends on DCFG.LPMCap.</p> <p>DCFG.LPMCap is 1'b0 - The core always responds with Timeout (that is, no response).</p> <p>DCFG.LPMCap is 1'b1 - The core responds with an ACK on successful LPM transaction, which requires that all of the following are satisfied:</p> <ol style="list-style-type: none"> 1. There are no PID or CRC5 errors in both the EXT token and the LPM token (if not true, inactivity results in a timeout ERROR). 2. No data is pending in the Transmit FIFO and OUT endpoints not in flow controlled state (else NYET). 3. The BESL value in the LPM token is less than or equal to LPM_NYET_thres[3:0] |
| 19 | RW | 0x0 | <p>KeepConnect KeepConnect When 1, this bit enables the save and restore programming model by preventing the core from disconnecting from the host when DCTL.RunStop is set to 0. It also enables the Hibernation Request Event to be generated when the link goes to U3 or L2. The device core disconnects from the host when DCTL.RunStop is set to 0.</p> <p>This bit indicates whether to preserve this behavior (0), or if the core must not disconnect when RunStop is set to 0 (1).</p> <p>This bit also prevents the LTSSM from automatically going to U0/L0 when the host requests resume from U3/L2.</p> |
| 18 | RW | 0x0 | <p>L1HibernationEn L1HibernationEn When this bit is set along with KeepConnect, the device core generates a Hibernation Request Event if L1 is enabled and the HIRD value in the LPM token is larger than the threshold programmed in DCTL.HIRD_Thres.</p> <p>The core does not exit the LPM L1 state until software writes Recovery into the DCTL.ULStChngReq field.</p> <p>This prevents corner cases where the device is entering hibernation at the same time the host is attempting to exit L1.</p> |
| 17 | RW | 0x0 | <p>CRS Controller Restore State This command is similar to the USBCMD.CRS bit in host mode and initiates the restore process. When software sets this bit to 1, the controller immediately sets DSTS.RSS to 1. When the controller has finished the restore process, it sets DSTS.RSS to 0.</p> <p>Note: When read, this field always returns 0.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | RW | 0x0 | CSS Controller Save State This command is similar to the USBCMD.CSS bit in host mode and initiates the save process. When software sets this bit to 1, the controller immediately sets DSTS.SSS to 1. When the controller has finished the save process, it sets DSTS.SSS to 0. Note: When read, this field always returns 0. |
| 15:13 | RO | 0x0 | reserved |
| 12 | RW | 0x0 | INITU2ENA Initiate U2 Enable 1'b0: May not initiate U2 (default) 1'b1: May initiate U2 On USB reset, hardware clears this bit to 0. Software sets this bit after receiving SetFeature(U2_ENABLE), and clears this bit when ClearFeature(U2_ENABLE) is received. If DCTL[11] (AcceptU2Ena) is 0, the link immediately exits U2 state. |
| 11 | RW | 0x0 | ACCEPTU2ENA Accept U2 Enable 1'b0: Reject U2 except when Force_LinkPM_Accept bit is set (default) 1'b1: Core accepts transition to U2 state if nothing is pending on the application side. On USB reset, hardware clears this bit to 0. Software sets this bit after receiving a SetConfiguration command. |
| 10 | RW | 0x0 | INITU1ENA Initiate U1 Enable 1'b0: May not initiate U1 (default); 1'b1: May initiate U1. On USB reset, hardware clears this bit to 0. Software sets this bit after receiving SetFeature(U1_ENABLE), and clears this bit when ClearFeature(U1_ENABLE) is received. If DCTL[9] (AcceptU1Ena) is 0, the link immediately exits U1 state. |
| 9 | RW | 0x0 | ACCEPTU1ENA Accept U1 Enable 1'b0: Core rejects U1 except when Force_LinkPM_Accept bit is set (default) 1'b1: Core accepts transition to U1 state if nothing is pending on the application side. On USB reset, hardware clears this bit to 0. Software sets this bit after receiving a SetConfiguration command. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 8:5 | RW | 0x0 | <p>ULSTCHNGREQ ULSTCHNGREQ</p> <p>Software writes this field to issue a USB/Link state change request. A change in this field indicates a new request to the core.</p> <p>If software wants to issue the same request back-to-back, it must write a 0 to this field between the two requests. The result of the state change request is reflected in the USB/Link State in DSTS. These bits are self-cleared on the MAC Layer exiting suspended state.</p> <p>If software is updating other fields of the DCTL register and not intending to force any link state change, then it must write a 0 to this field.</p> <p>SS Compliance mode is normally entered and controlled by the remote link partner. Refer to the USB 3.0 specification.</p> <p>Alternatively, you can force the local link directly into compliance mode, by resetting the SS link with the RUN/STOP bit set to zero. If you then write 10 to the USB/Link State Change field and 1 to RUN/STOP, the link goes to compliance mode.</p> <p>Once you are in compliance, you may alternately write zero and 10 to this field to advance the compliance pattern.</p> <p>In SS mode:</p> <p>Value: Requested Link State Transition/Action</p> <ul style="list-style-type: none"> 0: No Action 4: SS.Disabled 5: Rx.Detect 6: SS.Inactive 8: Recovery 10: Compliance Others: Reserved <p>In HS/FS/LS mode:</p> <p>Value: Requested USB state transition</p> <ul style="list-style-type: none"> 8: Remote wakeup request Others: Reserved <p>The Remote wakeup request must be issued 2us after the device goes into suspend state (DSTS[21:18] is 3).</p> <p>Note: After coming out of hibernation, software must write 8 (Recovery) into this field to confirm exit from the suspended state.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 4:1 | RW | 0x0 | TSTCTL Test Control 4'b000: Test mode disabled 4'b001: Test_J mode 4'b010: Test_K mode 4'b011: Test_SE0_NAK mode 4'b100: Test_Packet mode 4'b101: Test_Force_Enable Others: Reserved |
| 0 | RO | 0x0 | reserved |

USB3_DEVEN

Address: Operational Base + offset (0xc708)

Device Event Enable Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | reserved |
| 12 | RW | 0x0 | VENDEVTSTRCVDEN Vendor Device Test LMP Received Event 1: Enable this event 0: Disable this event |
| 11:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | ERRTICERREVREN Erratic Error Event Enable 1: Enable this event 0: Disable this event |
| 8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | SOFTEVTEN Start of (u)frame Event Enable 1: Enable this event 0: Disable this event |
| 6 | RW | 0x0 | U3L2L1SuspEn U3/L2-L1 Suspend Event Enable 1: Enable this event 0: Disable this event |
| 5 | RW | 0x0 | HibernationReqEvtEn Hibernation Request Event Enable 1: Enable this event 0: Disable this event |
| 4 | RW | 0x0 | WKUPEVTEN Resume/Remote Wakeup Detected Event Enable 1: Enable this event 0: Disable this event |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 3 | RW | 0x0 | ULSTCNGEN USB/Link State Change Event Enable 1: Enable this event 0: Disable this event |
| 2 | RW | 0x0 | CONNECTDONEEVTEN Connection Done Event Enable 1: Enable this event 0: Disable this event |
| 1 | RW | 0x0 | USBRSTEVTEN USB Reset Event Enable 1: Enable this event 0: Disable this event |
| 0 | RW | 0x0 | DISSCONNEVTEN Disconnect Detected Event Enable 1: Enable this event 0: Disable this event |

USB3_DSTS

Address: Operational Base + offset (0xc70c)

Device Status Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:30 | RO | 0x0 | reserved |
| 29 | RO | 0x0 | DCNRD Device Controller Not Ready The bit indicates that the core is in the process of completing the state transitions after exiting from hibernation. To complete the state transitions, it takes 256 bus clock cycles from the time DCTL[31].Run/Stop is set. During hibernation, if the UTMI/ULPI PHY is in suspended state, then the 256-bus clock cycle delay starts after the PHY exited suspended state. Software must set DCTL[31].Run/Stop to 1 and wait for this bit to be de-asserted to zero before processing DSTS.USBLnkSt. |
| 28:26 | RO | 0x0 | reserved |
| 25 | RO | 0x0 | RSS RSS Restore State Status This bit is similar to the USBSTS.RSS in host mode. When the controller finishes the restore process, it completes the command by setting DSTS.RSS to 0. |
| 24 | RO | 0x0 | SSS SSS Save State Status This bit is similar to the USBSTS.SSS in host mode. When the controller has finished the save process, it completes the command by setting DSTS.SSS to 0. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 23 | RO | 0x0 | <p>COREIDLE Core Idle</p> <p>The bit indicates that the core finished transferring all RxFIFO data to system memory, writing out all completed descriptors, and all Event Counts are zero.</p> <p>Note: While testing for Reset values, mask out the read value. This bit represents the changing state of the core and does not hold a static value.</p> |
| 22 | RO | 0x1 | <p>DEVCTRLHLT Device Controller Halted</p> <p>This bit is set to 0 when the Run/Stop bit in the DCTL register is set to 1.</p> <p>The core sets this bit to 1 when, after SW sets Run/Stop to 0, the core is idle and the lower layer finishes the disconnect process. When Halted=1, the core does not generate Device events.</p> <p>Note: The core does not set this bit to 1 if GEVNTCOUNTn has some valid value. Software needs to acknowledge the events that are generated (by writing to GEVNTCOUNTn) while it is waiting for this bit to be set to 1.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 21:18 | RO | 0x4 | <p>USBLNKST USBLNKST SS mode: LTSSM State 4'h0: U0 4'h1: U1 4'h2: U2 4'h3: U3 4'h5: RX_DET 4'h6: SS_INACT 4'h7: POLL 4'h8: RECOV 4'h9: HRESET 4'ha: CMPLY 4'hb: LPBK 4'hf: Resume/Reset In HS/FS/LS mode: 4'h0: On state 4'h2: Sleep (L1) state 4'h3: Suspend (L2) state 4'h4: Disconnected state (Default state) 4'h5: Early Suspend state (valid only when Hibernation is disabled, GCTL[1].GblHibernationEn = 0) 4'he: Reset (valid only when Hibernation is enabled, GCTL[1].GblHibernationEn = 1) 4'hf: Resume (valid only when Hibernation is enabled, GCTL[1].GblHibernationEn = 1) The link state Resume/Reset indicates that the core received a resume or USB reset request from the host while the link was in hibernation. Software must write 8 (Recovery) to the DCTL.ULStChngReq field to acknowledge the resume/reset request. When Hibernation is enabled, GCTL[1].GblHibernationEn = 1, this field USBLnkSt is valid only when DCTL[31].Run/Stop set to 1 and DSTS[29].DCNRD = 0.</p> |
| 17:3 | RO | 0x0 | reserved |
| 2:0 | RU | 0x4 | <p>CONNECTSPD Connected Speed Indicates the speed at which the DWC_usb3 core has come up after speed detection through a chirp sequence. 3'b100: SuperSpeed (PHY clock is running at 125 or 250 MHz) 3'b000: High-speed (PHY clock is running at 30 or 60 MHz) 3'b001: Full-speed (PHY clock is running at 30 or 60 MHz)</p> |

USB3_DGCMDPAR

Address: Operational Base + offset (0xc710)

Device Generic Command Parameter Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | PARAMETER PARAMETER This register indicates the device command parameter. This must be programmed before or along with the device command. The available device commands are listed in DGCMD register. |

USB3_DGCMD

Address: Operational Base + offset (0xc714)

Device Generic Command Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15:12 | RO | 0x0 | CMDSTATUS Command Status 1: CmdErr: Indicates that the device controller encountered an error while processing the command. 0: Indicates command success |
| 11 | RO | 0x0 | reserved |
| 10 | R/W SC | 0x0 | CMDACT Command Active The software sets this bit to 1 to enable the device controller to execute the generic command. The device controller sets this bit to 0 after executing the command. |
| 9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | CMDIOC Command Interrupt on Complete When this bit is set, the device controller issues a Generic Command Completion event after executing the command. Note that this interrupt is mapped to DCFG.IntrNum. Note: This field must not set to 1 if the DCTL.RunStop field is 0. |
| 7:0 | RW | 0x00 | CMDTYP Command Type Specifies the type of command the software driver is requesting the core to perform. 00h: Reserved 01h: Set Endpoint Configuration - 64 or 96-bit Parameter 02h: Set Endpoint Transfer Resource Configuration - 32-bit Parameter 03h: Get Endpoint State - No Parameter Needed 05h: Clear Stall (see Set Stall) - No Parameter Needed 06h: Start Transfer - 64-bit Parameter 07h: Update Transfer - No Parameter Needed 08h: End Transfer - No Parameter Needed 09h: Start New Configuration - No Parameter Needed |

USB3_DALEPENA

Address: Operational Base + offset (0xc720)

Device Active USB Endpoint Enable Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>USBACTEP USB Active Endpoints This field indicates if a USB endpoint is active in the current configuration and interface. It applies to USB IN endpoints 0~15 and OUT endpoints 0~15, with one bit for each of the 32 possible endpoints. Even numbers are for USB OUT endpoints, and odd numbers are for USB IN endpoints, as follows: Bit[0]: USB EP0-OUT Bit[1]: USB EP0-IN Bit[2]: USB EP1-OUT Bit[3]: USB EP1-IN ... The entity programming this register must set bits 0 and 1 because they enable control endpoints that map to physical endpoints (resources) after USBReset. Hardware clears these bits for all endpoints (other than EP0-OUT and EP0-IN) after detecting a USB reset event. After receiving SetConfiguration and SetInterface requests, the application must program endpoint registers accordingly and set these bits. For more information, Pls see "Flexible Endpoint Mapping" section.</p> |

USB3_DEPnCMDPAR2

Address: Operational Base + offset (0xc800 + 4*n), n=0~12

Device Physical Endpoint-n Command Parameter 2 Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>PARAMETER PARAMETER This register indicates the physical endpoint command Parameter 2. It must be programmed before issuing the command.</p> |

USB3_DEPnCMDPAR1

Address: Operational Base + offset (0xc804 + 4*n), n=0~12

Device Physical Endpoint-n Command Parameter 1 Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | <p>PARAMETER PARAMETER This register indicates the physical endpoint command Parameter 1. It must be programmed before issuing the command.</p> |

USB3_DEPnCMDPARO

Address: Operational Base + offset (0xc808 + 4*n), n=0~12

Device Physical Endpoint-n Command Parameter 0 Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | PARAMETER PARAMETER This register indicates the physical endpoint command Parameter 0. It must be programmed before issuing the command. |

USB3_DEPnCMD

Address: Operational Base + offset (0xc80c + 4*n), n=0~12

Device Physical Endpoint-n Command Register

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | COMMANDPARAM Command Parameters or Event Parameters when this register is written: For Start Transfer command: The 16-bit StreamID assigned to this transfer For Start Transfer command applied to an isochronous endpoint: StartMicroFramNum, Indicates the (micro)frame number to which the first TRB applies. For Update Transfer, End Transfer, and Start New Configuration commands: [22:16]: Transfer Resource Index (XferRscIdx). The hardware-assigned transfer resource index for the transfer, which was returned in response to the Start Transfer command. The application software-assigned transfer resource index for a Start New Configuration command. Event Parameters (EventParam), when this register is read. Please see bits [31:16] in the "Device Endpoint-n Events: DEPEVT" of the Databook. |
| 15:12 | RW | 0x0 | CMDSTATUS Command Completion Status The information is in the same format as bits 15:12 of the Endpoint Command Complete event, Please see "Device Endpoint-n Events: DEPEVT" in the Databook. |
| 11 | RW | 0x0 | HIPRI_FORCERM HighPriority/ForceRM HighPriority: Only valid for Start Transfer command ForceRM: Only valid for End Transfer command ClearPendIN: Only valid for Clear Stall command . Software sets this bit to clear any pending IN transaction (on that endpoint) stuck at the lower layers when a Clear Stall command is issued. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 10 | RW | 0x0 | CMDACT Command Active Software sets this bit to 1 to enable the device endpoint controller to execute the generic command. The device controller sets this bit to 0 when the CmdStatus field is valid and the endpoint is ready to accept another command. This does not imply that all the effects of the previously-issued command have taken place. |
| 9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | CMDIOC Command Interrupt on Complete When this bit is set, the device controller issues a generic Endpoint Command Complete event after executing the command. Note that this interrupt is mapped to DEPCFG.IntrNum. When the DEPCFG command is executed, the command interrupt on completion goes to the interrupt pointed by the DEPCFG.IntrNum in the current command. Note: This field must not set to 1 if the DCTL.RunStop field is 0. |
| 7:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | CMDTYP Command Type Specifies the type of command the software driver is requesting the core to perform. 00h: Reserved 01h: Set Endpoint Configuration - -64 or 96-bit Parameter 02h: Set Endpoint Transfer Resource Configuration - 32-bit parameter 03h: Get Endpoint State - No Parameter Needed 04h: Set Stall - No Parameter Needed 05h: Clear Stall (see Set Stall) - No Parameter Needed 06h: Start Transfer - 64-bit Parameter 07h: Update Transfer - No Parameter Needed 08h: End Transfer - No Parameter Needed 09h: Start New Configuration - No Parameter Needed |

9.5 Interface Description

Table 9-2 USB2.0 PHY and PCIe&SS PHY Interface Description

| Module Pin | Direction | Pin Name | Descriptions |
|------------|-----------|--------------|--------------------------|
| USB0ID | I/O | USB_OTG_ID | USB2.0 PHY OTG Port ID |
| USB0PN | I/O | USB_OTG_DM | USB2.0 PHY OTG Port PN |
| USB0PP | I/O | USB_OTG_DP | USB2.0 PHY OTG Port PP |
| VBUS | I/O | USB_OTG_VBUS | USB2.0 PHY OTG Port VBUS |
| TXP_A | I/O | PCIE_TX0P | |
| TXN_A | I/O | PCIE_TX0N | |

| Module Pin | Direction | Pin Name | Descriptions |
|------------|-----------|------------|--------------|
| RXP_A | I/O | PCIE_RX0P | |
| RXN_A | I/O | PCIE_RX0N | |
| RBIAS | I/O | PCIE_RBIAS | |

9.6 Application Notes

9.6.1 Some Special Settings before Initialization

Set USB3.0 OTG controller AXI master secure setting.

Clear USB2.0 only mode setting (bit 0 of register GRF_PIPE_CON3 in Chapter GRF)

Set PHYIF to 1 to use 16-bit UTMI+ interface (see register GUSB2PHYCFG0)

Clear ENBLSLPM to 0 to disable sleep and I1 suspend (see register GUSB2PHYCFG0)

Clear U2_FREECLK_EXITSTS to 0 (see register GUSB2PHYCFG0)

Clear DEV_FORCE_20_CLK_FOR_30_CLK to 0 (see register GUCTL1)

Clear DELAYP1TRANS to 0 (see register GUSB3PIPECTL0)

Hold USB3.0 controller in resetting during USB3PHY initialization

9.6.2 OTG Programming Model

When detect ID change event (see USBPHY_GRF) or VBUS change event (see USBPHY_GRF) after disconnect, it means a USB3/2 OTG A device or B device may connect, then check status of ID(see USBPHY_GRF). If ID==0, it will work as A device, then follow host programming flow; if ID==1, it will work as B device, then it follow device programming flow.

Note: USB3.0 OTG doesn't support host/device mode swapping through HNP and RSP.

Note: PCIe&SS PHY is combo with PCIe PHY and Super-Speed PHY, when PCIe function is on, USB3.0 OTG can only work on usb2.0 mode.

Chapter 10 PCIe Controller

10.1 Overview

The PCI Express(PCIE) is a high performance, general purpose I/O interconnect defined for a wide variety of future computing and communication platforms. The PCIe designed to be used as a general purpose serial I/O interconnect in multiple market segments, including desktop, mobile, server, storage and embedded communications. It is compliant with PCI Express Specifications 1.1, 2.1.

PCIe supports the following features:

- Compatible with PCI Express Base Specification Revision 2.1
- Dual operation mode: Root Complex(RC)and End Point(EP)
- Maximum link width is 2, single bi-directional Link interface
- Maximum Payload Size of 128 bytes
- Maximum 32 Non-Posted outstanding transactions
- Support 2.5GT/s and 5.0 GT/s serial data transmission rate per lane per direction
- Support 100MHz differential clock output (optional with SSC) for system application
- Embedded DMA with Hardware Flow Control
- Support Latency Tolerance Reporting(LTR)
- Support Optimized Buffer Flush and Fill(OBFF)
- Support Resizable BAR Capability
- Support Single Physical PCI Function in Endpoint Mode
- Support Legacy Interrupt
- Support MSI with Per-Vector Masking (PVM) and 32 multiple MSI
- Support MSI-X with Per-Vector Masking (PVM) and 64 MSI-X table size
- Support ECRC Generation and Checking
- Support Outbound and Inbound address translation
- Support Dynamic Power Allocation Capability(DPA)
- Support PCI Express Active State Power Management (ASPM) state L0s and L1
- Support L1 Power Management Substate
- Support PCI Function power states D0, D1 and D3, and the corresponding link power states L0, L1 and L2
- Support RAS DES (Debug, Error Injection, and Statistics)
- Support Automatic Lane Reversal
- Support PCI Express Advanced Error Reporting (AER) with Header Logging

10.2 Block Diagram

PCIe comprises with:

- APB: APB Interface
- AXI_S: AXI Slave Interface
- AXI_M: AXI Master Interface
- AXI_DBI: AXI Slave Interface for configuration
- Client: Client Logic
- CLKRST: Clock and Reset related logic
- Memory: SRAMs and Buffers
- AMBA Bridge: AXI to Core Bridge
- EDMA: Embedded DMA
- PCIe Core: PCIe Controller Core
- PM: Power Management Unit
- PCIe 2.0 USB 3.0 Combo PHY

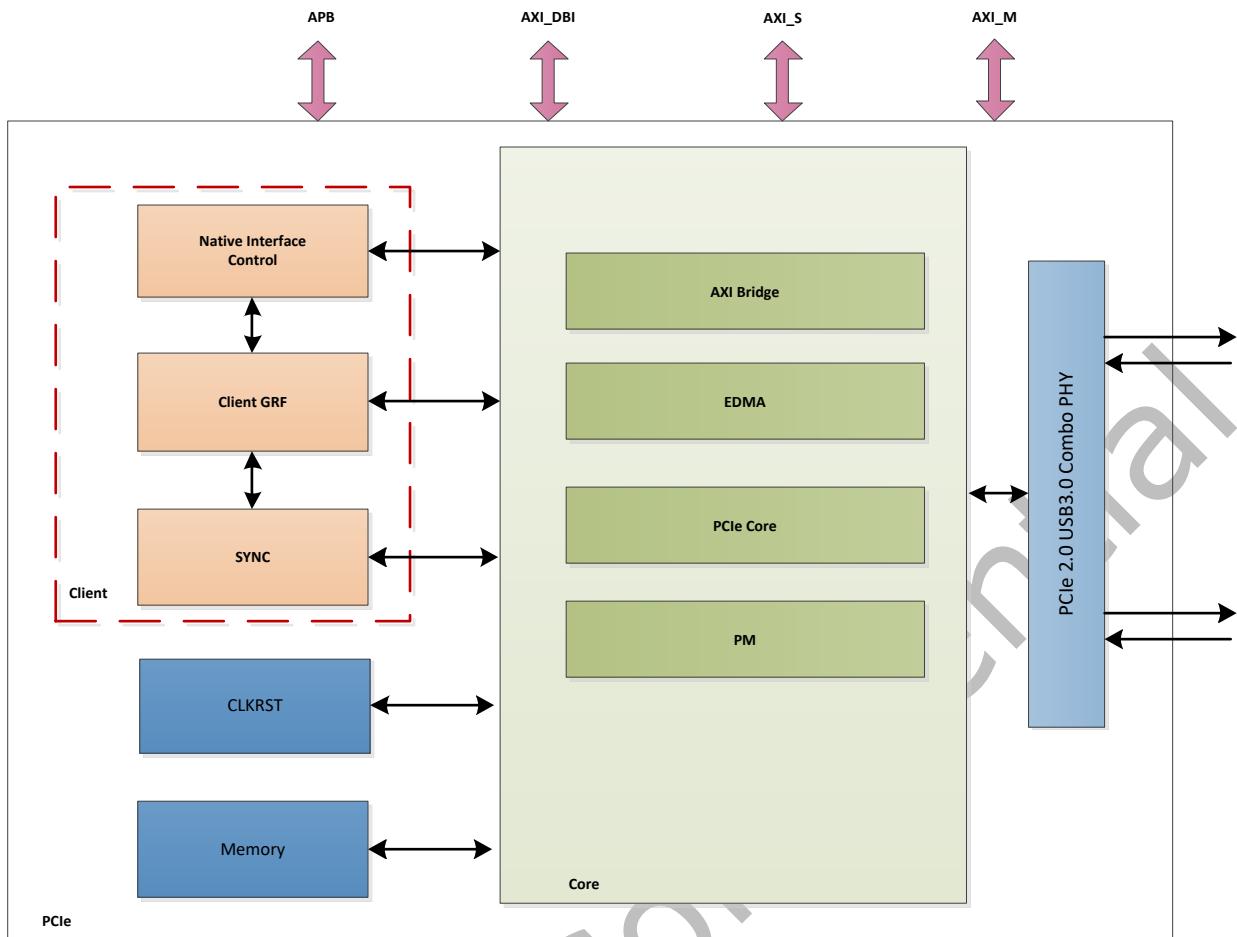


Fig. 10-1 PCIe Block Diagram

Block function description will be discussed in next section.

10.3 Function Description

10.3.1 Application Interface and AXI Bridge

Application Interface comprises with four standard interface: one AXI master interface, two AXI slave interface and one APB interface. These interface bridging capability for directly adding a PCI Express link to an AXI system fabric.

The AXI bridge module acts as a bridge between the standard AXI interfaces and the PCIe controller native interfaces. The bridge interconnects the AXI interfaces within an AXI-embedded system with a remote PCIe link, as either a root complex port or as an endpoint port.

The AXI master interfaces enable a remote PCIe device to read and write to an AXI slave connected to the AXI bridge. The AXI link slave interface enables an AXI master to read and write through the AXI bridge to a remote PCIe device. The DBI slave enables an AXI master to access the controller's registers. The APB enables Application to access the client registers.

10.3.2 Client Logic

Client Logic Consists of some additional logic used for application to interact with PCIe Controller. For example, applications can access client registers to send/receive PCIe Message, request to enter/exit PM state, deal with interrupts, configure some basic operation mode, read some basic debug information, and so on. For more details, please refer to client registers description and application notes for more information.

10.3.3 Embedded DMA

The RC system CPU, or the EP application CPU, can offload the transferring of large blocks of data to the embedded DMA controller, leaving the CPU free to perform other tasks. The

embedded DMA have one read channel and one write channel. It can simultaneously perform the two types of memory transactions:

DMA write: Transfer (copy) of a block of data from local (application) memory to remote (link partner) memory.

DMA read: Transfer (copy) of a block of data from remote (link partner) memory to local (application) memory.

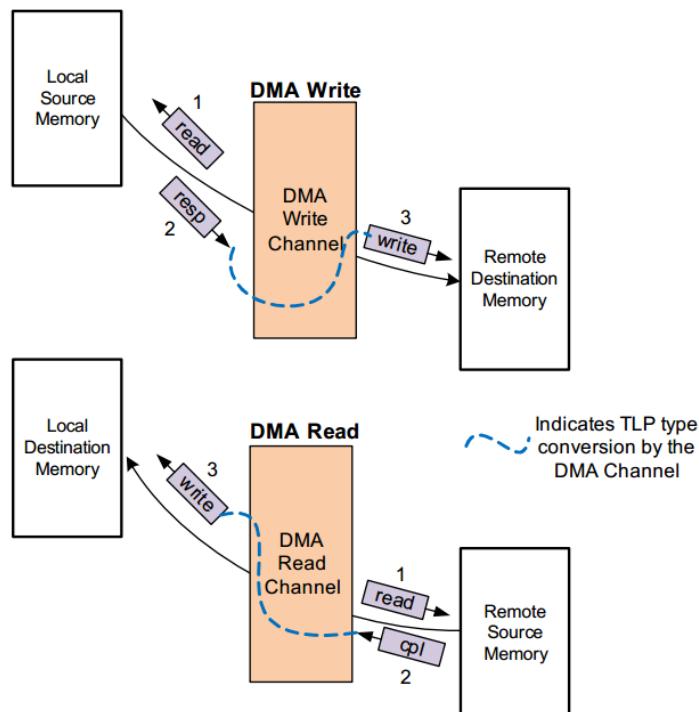


Fig. 10-2 System Level View of PCIe EDMA

Therefore the EDMA supports full duplex operation, processing read and write transfers at the same time, and in parallel with normal (non-DMA) traffic. Upon completion of a DMA transfer or an error, the DMA optionally interrupts the local CPU or sends an interrupt MWr (Memory Write) to the remote CPU. The DMA is highly configurable and can be programmed by using the local DBI AXI Slave interface.

The EDMA provides a linked list (LL) mode to efficiently move data from source to destination with minimal intervention from the local CPU. For details, please refer to DMA Registers Description and Application notes for more information.

10.3.4 PCIe Core

PCIe Core deal with the PCIe protocol, and consists of three main module:

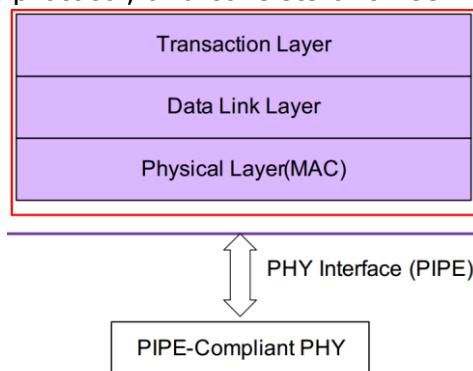


Fig. 10-3 PCIe Core Architecture Overview

The PCIe Core implements the basic functionality for the PCI Express physical, link, and transaction layers. This module implements a large part of the transaction layer logic, all of the data link layer logic, and the MAC portion of the physical layer, including the link training and status state machine (LTSSM). The Core connects to the external PHY through the standard PIPE interface.

10.3.5 Power Management

PCIe Controller support PCIe ASPM, L1 Substate and PCI PM. For proper understanding of PCIe Power Management, you should be familiar with Chapter 5, Power Management of the PCI Express Base Specification, and PCI-SIG Engineering Change Notice ECN L1 Substates with CLKREQ#. For application details, please refer to Application Notes for more information.

10.3.6 PCIe 2.0 PHY

PCIe Controller shares a combo PHY with USB 3.0 Controller. The PCIe PHY, which consists of Physical Coding Sub-layer(PCS) and Physical Media Attachment Layer(PMA), includes all circuitry for interface operation, including 8/10 encoding/decoding, driver and input buffers, parallel-to-serial and serial-to-parallel conversion, PLL(s), and impedance matching circuitry. The PHY provides standard PIPE interface with the Media Access Layer for exchanging information. This PHY is responsible for converting information received from the PCIe Core into an appropriate serialized format and transmitting it across the PCIe Link at a frequency and width compatible with the device connected to the other side of the Link.

10.4 Interface Description

Table 10-1 PCIe Interface Description

| Module Pin | Direction | Pin Name | IOMUX Setting |
|----------------------|-----------|--------------------------------------|---|
| RXN_A | I | PCIE_RX0N | NS |
| RXP_A | I | PCIE_RX0P | NS |
| TXN_A | O | PCIE_TX0N | NS |
| TXP_A | O | PCIE_TX0P | NS |
| RXN_B | I | PCIE_RX1N | NS |
| RXP_B | I | PCIE_RX1P | NS |
| TXN_B | O | PCIE_TX1N | NS |
| TXP_B | O | PCIE_TX1P | NS |
| PCIE_REFCLKP | I/O | PCIE_REFCLKP | NS |
| PCIE_REFCLKN | I/O | PCIE_REFCLKN | NS |
| RBIAS | I/O | PCIE_RBIAS | NS |
| pcie_button_rst_n | I | GPIO0_A2/PCIE_BUTTONRST | PMUGRF_GPIO0A_IOMUX[5:4]=2'b01 SGRF_SOC_CON1[13]=1'b1 |
| pcie_perst_n | I | GPIO0_A5/PCIE_PERST_M0 | PMUGRF_GPIO0A_IOMUX[11:10]=2'b01 PMUGRF_SOC_CON0[8]=1'b1 |
| | | GPIO0_B6/PCIE_PERST_M1 | PMUGRF_SOC_CON0[8]=1'b0 |
| pcie_wake_in/out | I/O | GPIO0_A7/PCIE_WAKE_M0 | PMUGRF_GPIO0A_IOMUX[15:14]=2'b01 PMUGRF_SOC_CON0[9]=1'b1 |
| | | GPIO0_C5/PCIE_WAKE_M1/PWM2 | PMUGRF_GPIO0C_P[11:10]=2'b01 PMUGRF_SOC_CON0[9]=1'b0 |
| pcie_clkreq_in/out_n | I/O | GPIO0_A3/PCIE_CLKREQN_M0/SDMMC0_DETN | PMUGRF_GPIO0A_IOMUX[7:6]=2'b10 PMUGRF_SOC_CON0[10]=1'b1 |

| Module Pin | Direction | Pin Name | IOMUX Setting |
|------------|-----------|------------------------------------|--|
| | | GPIO0_C6/PCIE_CLKREQN_M1/UART3_CTS | PMUGRF_GPIO0C_P[13:12]=2'b01 PMUGRF_SOC_CON0[10]=1'b0 |

Notes: I=input, O=output, I/O=input/output, bidirectional

10.5 Register Description

PCIe Registers can be accessed by either local CPU through AXI Dbi bus or remote RC device through PCIe link or both. The way through local CPU will be mentioned as 'Dbi' and the way through PCIe link will be mentioned as 'wire' in this section.

Some read-only registers can be made temporarily R/W when you write 1 to the DBI_RO_WR_EN bit of the MISC_CONTROL_1_OFF register. These registers will be specifically mentioned in their description if they have this feature.

10.5.1 Internal Address Mapping

PCIe local address mapping is discussed in this section.

Table 10-2 PCIe Client and Core Register Address Mapping

| Base Address | Device | Address Length | Offset Address Range |
|--------------|----------------------|----------------|----------------------|
| 0xFC000000 | Core Register | 4M BYTE | 0x00000 ~ 0x3fffff |
| 0xFC400000 | Client Register | 64K BYTE | 0x0000~0xffff |
| 0xF8000000 | PCIe Outbound Memory | 64M BYTE | 0x0000000~0x3fffff |
| 0xFF380000 | PHY Register | 64K BYTE | 0x0000~0xffff |

Core Register map is as follows:

RC mode:

Table 10-3 PCIe Core Register Map in RC mode

| Offset Address Range | Register Block | Description |
|----------------------|----------------|---|
| 0x0000~0x003c | TYPE1_HDR | PCI-Compatible Configuration Space Header Type1 |
| 0x0040~0x0044 | PM_CAP | PCI Power Management Capability Structure |
| 0x0050~0x0064 | MSI_CAP | MSI Capability Structure |
| 0x0070~0x00a0 | PCIE_CAP | PCI Express Capability Structure |
| 0x00b0~0x00b8 | MSIX_CAP | MSI-X Capability Structure |
| 0x0100~0x0144 | AER_CAP | Advanced Error Reporting Capability Structure |
| 0x0148~0x0154 | SPCIE_CAP | Secondary PCI Express Capability Structure |
| 0x0160~0x016c | L1SUB_CAP | L1 Substates Capability Structure |
| 0x01a0~0x0288 | RAS DES CAP | RAS D.E.S. Capability Structure |
| 0x0700~0x0b90 | PORT_LOGIC | Port Logic |
| 0x1000b0 | MSIX_CAP_DB12 | Shadow Block: MSI-X Capability Structure |
| 0x300000~0x301f14 | ATU_CAP | ATU Por Logic Structure |
| 0x380000~0x380320 | DMA_CAP | DMA Port Logic Structure |

EP mode:

Table 10-4 PCIe Core Register Map in EP mode

| Offset Address Range | Register Block | Description |
|----------------------|----------------|------------------------------|
| 0x0000~0x003c | TYPE0_HDR | PCI-Compatible Configuration |

| Offset Address Range | Register Block | Description |
|----------------------|----------------------|--|
| | | Space Header Type0 |
| 0x0040~0x0044 | <u>PM CAP</u> | PCI Power Management Capability Structure |
| 0x0050~0x0064 | <u>MSI CAP</u> | MSI Capability Structure |
| 0x0070~0x00a0 | <u>PCIE CAP</u> | PCI Express Capability Structure |
| 0x00b0~0x00b8 | <u>MSIX CAP</u> | MSI-X Capability Structure |
| 0x0100~0x0144 | <u>AER CAP</u> | Advanced Error Reporting Capability Structure |
| 0x0148~0x0154 | <u>SPCIE CAP</u> | Secondary PCI Express Capability Structure |
| 0x0158~0x015c | <u>LTR CAP</u> | Latency Tolerance Reporting Capability Structure |
| 0x0160~0x016c | <u>L1SUB CAP</u> | L1 Substates Capability Structure |
| 0x0170~0x0184 | <u>DPA CAP</u> | DPA Capability Structure |
| 0x01a0~0x0288 | <u>RAS DES CAP</u> | RAS D.E.S. Capability Structure |
| 0x02a0~0x02d0 | <u>RESBAR CAP</u> | Resizable BAR Capability Structure |
| 0x0700~0x0b90 | <u>PORT LOGIC</u> | Port Logic |
| 0x1000b0 | <u>MSIX CAP DBI2</u> | Shadow Block: MSI-X Capability |
| 0x300000~0x301f14 | <u>ATU CAP</u> | ATU Port Logic Structure |
| 0x380000~0x380320 | <u>DMA CAP</u> | DMA Port Logic Structure |

10.5.2 PCIe Client Registers Summary

Client Registers provide some control, status, and debug interface to PCIe controller. Operations like general control, interrupt, power management can be handled by client registers. Client Registers can only be accessed by local CPU.

| Name | Offset | Size | Reset Value | Description |
|--|--------|------|-------------|---|
| <u>PCIE CLIENT GENERAL C ON</u> | 0x0000 | W | 0x00000000 | General Control register |
| <u>PCIE CLIENT INTR STAT US MSG RX</u> | 0x0004 | W | 0x00000000 | Interrupt status register related to message reception. |
| <u>PCIE CLIENT INTR STAT US LEGACY</u> | 0x0008 | W | 0x00000000 | Interrupt status register related to legacy interrupt. |
| <u>PCIE CLIENT INTR STAT US ERR</u> | 0x000c | W | 0x00000000 | Interrupt status register related to error detection. |
| <u>PCIE CLIENT INTR STAT US MISC</u> | 0x0010 | W | 0x00000000 | Interrupt status register related to miscellaneous operation. |
| <u>PCIE CLIENT INTR STAT US PMC</u> | 0x0014 | W | 0x00000000 | Interrupt status register related power management control. |
| <u>PCIE CLIENT INTR MASK MSG RX</u> | 0x0018 | W | 0x00007737 | Interrupt mask register related to message reception. |
| <u>PCIE CLIENT INTR MASK LEGACY</u> | 0x001c | W | 0x000000ff | Interrupt mask register related to legacy interrupt. |
| <u>PCIE CLIENT INTR MASK ERR</u> | 0x0020 | W | 0x0000177f | Interrupt mask register related to error detection. |
| <u>PCIE CLIENT INTR MASK MISC</u> | 0x0024 | W | 0x00007fff | Interrupt mask register related to miscellaneous operation. |
| <u>PCIE CLIENT INTR MASK PMC</u> | 0x0028 | W | 0x000001ff | Interrupt mask register related power management control. |

| Name | Offset | Size | Reset Value | Description |
|----------------------------------|--------|------|-------------|--|
| PCIE CLIENT POWER CON | 0x002c | W | 0x00000008 | Power management control register. |
| PCIE CLIENT POWER STATUS | 0x0030 | W | 0x00000000 | Power management status register. |
| PCIE CLIENT MSG GEN CON | 0x0034 | W | 0x00000000 | Message generation control register |
| PCIE CLIENT MSI GEN CON | 0x0038 | W | 0x00000000 | MSI generation control register |
| PCIE CLIENT MSI GEN FNUM TC | 0x0040 | W | 0x00000000 | MSI function number and TC set register |
| PCIE CLIENT RBAR SIZE INFO0 | 0x0044 | W | 0x00000000 | Resizable BAR2 to BAR0 size information register |
| PCIE CLIENT RBAR SIZE INFO1 | 0x0048 | W | 0x00000000 | Resizable BAR5 to BAR3 size information register |
| PCIE CLIENT DMA HSHAKE TOGG | 0x004c | W | 0x00000000 | DMA handshake toggle register |
| PCIE CLIENT VEN MSG RX INFO0 | 0x0050 | W | 0x00000000 | Register that contains Header information of vendor message that controller received |
| PCIE CLIENT VEN MSG RX INFO1 | 0x0054 | W | 0x00000000 | Register that contains Header information of vendor message that controller received |
| PCIE CLIENT VEN MSG RX INFO2 | 0x0058 | W | 0x00000000 | Register that contains Header information of vendor message that controller received |
| PCIE CLIENT VEN MSG TX CFG0 | 0x005c | W | 0x00000000 | Register that contains Header information of vendor message. all VEN_MSG_TX_INFOx register should not be changed before last Request has been granted. |
| PCIE CLIENT VEN MSG TX CFG1 | 0x0060 | W | 0x00000000 | Register that contains Header information of vendor message you want to send |
| PCIE CLIENT VEN MSG TX CFG2 | 0x0064 | W | 0x00000000 | Register that contains Header information of vendor message you want to send |
| PCIE CLIENT LTR MSG TX INFO | 0x0068 | W | 0x00000000 | Register that contains latency information of LTR message you want to send |
| PCIE CLIENT APP ERR REPORT INFO0 | 0x006c | W | 0x00000000 | Register that contains error information that application wants to report |

| Name | Offset | Size | Reset Value | Description |
|-----------------------------------|--------|------|-------------|---|
| PCIE CLIENT APP ERR_R PT_INFO1 | 0x0070 | W | 0x00000000 | Register that contains error information that application wants to report |
| PCIE CLIENT APP ERR_R PT_INFO2 | 0x0074 | W | 0x00000000 | Register that contains error information that application wants to report |
| PCIE CLIENT APP ERR_R PT_INFO3 | 0x0078 | W | 0x00000000 | Register that contains error information that application wants to report |
| PCIE CLIENT APP ERR_R PT_INFO4 | 0x007c | W | 0x00000000 | Register that contains error information that application wants to report |
| PCIE CLIENT OBFF WAKE_ELE_CFG | 0x0080 | W | 0xb4f05064 | Configure the pulse width and edge-to-edge timing of WAKE# in OBFF protocol. |
| PCIE CLIENT OBFF WAKE_DEBUG | 0x0084 | W | 0x00000000 | OBFF wake decoder debug register |
| PCIE CLIENT RX CPL_TIME_OUT_INFO | 0x0088 | W | 0x00000000 | Register that contains Header information of the timed out completion that expect to receive. |
| PCIE CLIENT TX CPL_TME_OUT_INFO | 0x008c | W | 0x00000000 | Register that contains Header information of the timed out completion that expect to send. |
| PCIE CLIENT TX CPL_TME_OUT_INFO2 | 0x0090 | W | 0x00000000 | Register that contains LUT information of the timed out completion that expect to send. |
| PCIE CLIENT LOCAL CRU_CTRL | 0x009c | W | 0x00000000 | GRF control signals to local CRU |
| PCIE CLIENT GENERAL DEBUG_CON | 0x0100 | W | 0x00000000 | General debug control register |
| PCIE CLIENT GENERAL DEBUG_INFO | 0x0104 | W | 0x00000000 | General debug information register |
| PCIE CLIENT SLC DEBUG_INFO_COMMON | 0x0108 | W | 0x00000000 | Common silicon debug information register |
| PCIE CLIENT SLC DEBUG_INFO_L0 | 0x010c | W | 0x00000000 | Lane 0 silicon debug information register |
| PCIE CLIENT SLC DEBUG_INFO_L1 | 0x0110 | W | 0x00000000 | Lane 1 silicon debug information register |
| PCIE CLIENT SLC DEBUG_INFO_V0 | 0x0114 | W | 0x00000000 | Virtual channel 0 silicon debug information register |
| PCIE CLIENT DIAG_STATUS_BUS_SEL | 0x0118 | W | 0x00000000 | Diagnostic Status bus select register |
| PCIE CLIENT DIAG_STATUS_BUS_INFO | 0x011c | W | 0x00000000 | Diagnostic Status bus information register |

| Name | Offset | Size | Reset Value | Description |
|---|--------|------|-------------|---|
| <u>PCIE CLIENT AXI MSTR MISC CON</u> | 0x0200 | W | 0x00000000 | AXI master sideband signals control register |
| <u>PCIE CLIENT AXI SLV A TU BYPASS</u> | 0x0204 | W | 0x00000000 | Address Translate Unit bypass set register |
| <u>PCIE CLIENT AXI SLV A WMISC HDR</u> | 0x0208 | W | 0x00000000 | AXI slave write address sideband signals register, provide TLP header information |
| <u>PCIE CLIENT AXI SLV A WMISC HDR3</u> | 0x020c | W | 0x00000000 | AXI slave write address sideband signals register, provide TLP header 3rd double word information |
| <u>PCIE CLIENT AXI SLV A WMISC HDR4</u> | 0x0210 | W | 0x00000000 | AXI slave write address sideband signals register, provide TLP header 4th double word information |
| <u>PCIE CLIENT AXI SLV A WMISC TAG</u> | 0x0214 | W | 0x00000000 | AXI Slave Write Request Tag register |
| <u>PCIE CLIENT AXI SLV M ISC INFO</u> | 0x0218 | W | 0x00000000 | AXI slave read address and write data sideband signals control register |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.3 PCIe Client Detail Register Description

PCIE CLIENT GENERAL CON

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | WO | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | link_req_rst_grt_sel Link down reset request grant control select: 1'b0: from pcie grf PCIE_CON0 bit 2; 1'b1: from client grf GENERAL_CON bit 3; |
| 12 | WO | 0x0 | app_err_repot Set 1 to report application error to controller. It is a self-clear bit to generate a pulse to controller. Application should provide error information in VEN_ERR_INFOx registers before setting this bit. |
| 11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | app_dbi_ro_wr_disable 1'b0: MISC_CONTROL_1_OFF.DBI_RO_WR_EN register field is read-write. 1'b1: MISC_CONTROL_1_OFF.DBI_RO_WR_EN register field is forced to 0 and is read-only |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9 | RW | 0x0 | <p>tx_lane_flip_en</p> <p>Performs manual lane reversal for transmit lanes. Used when automatic lane reversal does not occur.</p> <p>1'b0: Disable 1'b1: Enable</p> |
| 8 | RW | 0x0 | <p>rx_lane_flip_en</p> <p>Performs manual lane reversal for receive lanes. Used when automatic lane reversal does not occur.</p> <p>1'b0: Disable 1'b1: Enable</p> |
| 7:4 | RW | 0x0 | <p>device_type</p> <p>Device/port type. Indicates the specific type of this PCI Express function. It is also used to set the 'Device/Port Type' field of the 'PCI Express Capabilities Register'. The controller uses this input to determine the operating mode of the controller at run time.</p> <p>Defined encodings are:</p> <p>4'b0000: PCI Express endpoint 4'b0001: Legacy PCI Express endpoint 4'b0100: Root port of PCI Express root complex</p> |
| 3 | RW | 0x0 | <p>link_req_RST_grt</p> <p>Link down reset request grant control.</p> <p>1'b0: Disable link down reset request grant. 1'b1: Enable link down reset request grant.</p> |
| 2 | RW | 0x0 | <p>ltssm_enable</p> <p>Driven low by your application after cold, warm or hot reset to hold the LTSSM in the Detect state until your application is ready for the link training to begin. When your application has finished reprogramming the controller configuration registers using the DBI, it asserts app_ltssm_enable to allow the LTSSM to continue link establishment.</p> <p>Can also be used to delay hot resetting of the controller until you have read out any register status.</p> |
| 1 | RW | 0x0 | <p>app_req_retry_en</p> <p>Provides a capability to defer incoming configuration requests until initialization is complete. When app_req_retry_en is asserted, the controller completes incoming configuration requests with a configuration request retry status. Other incoming requests complete normally.</p> |
| 0 | RW | 0x0 | <p>app_init_RST</p> <p>Set 1 to send a hot reset to link partner. It is a self-clear bit to generate a hot reset pulse to controller. Only used in RC mode.</p> |

PCIE_CLIENT_INTR_STATUS_MSG_RX

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:15 | RO | 0x0 | reserved |
| 14 | W1 C | 0x0 | <p>obff_cpu_active_int Interrupt indicates that the controller received an 'CPU Active' OBFF message. 1'b0: No interrupt 1'b1: Interrupt EP only.</p> |
| 13 | W1 C | 0x0 | <p>obff_obff_int Interrupt indicates that the controller received an 'OBFF' OBFF message. 1'b0: No interrupt 1'b1: Interrupt EP only.</p> |
| 12 | W1 C | 0x0 | <p>obff_idle_int Interrupt indicates that the controller received an 'IDLE' OBFF message. 1'b0: No interrupt 1'b1: Interrupt EP only.</p> |
| 11 | RO | 0x0 | reserved |
| 10 | W1 C | 0x0 | <p>pm_turnoff_int Interrupt indicates that the controller received a PME Turnoff message. 1'b0: No interrupt 1'b1: Interrupt EP mode only.</p> |
| 9 | W1 C | 0x0 | <p>pm_to_ack_int Interrupt indicates that the controller received a PME_TO_Ack message. 1'b0: No interrupt 1'b1: Interrupt RC mode only.</p> |
| 8 | W1 C | 0x0 | <p>pm_pme_int Interrupt indicates that the controller received a PM_PME message. 1'b0: No interrupt 1'b1: Interrupt RC mode only.</p> |
| 7:6 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | W1C | 0x0 | pme_msi Asserted when all of the following conditions are true: 1. MSI or MSI-X is enabled. 2. The PME Interrupt Enable bit in Root Control register is set to 1. 3. The PME Status bit in Root Status register is set to 1. Note the difference between pme_msi and pme_int is that aer_rc_err_int is read only and rc_err_msi can be written 1 to clear. |
| 4 | RO | 0x0 | pme_int Asserted when all of the following conditions are true: The INTx Assertion Disable bit in the Command register is 0. The PME Interrupt Enable bit in the Root Control register is set to 1. The PME Status bit in the Root Status register is set to 1. This signal is used when MSI/MSI-X is NOT enabled; otherwise see cfg_aer_rc_err_msi. |
| 3 | RO | 0x0 | reserved |
| 2 | W1C | 0x0 | ltr_msg_int Interrupt that indicates the controller received a LTR message. The LTR message information is available in PL_LTR_LATENCY_OFF registers. 1'b0: No interrupt 1'b1: Interrupt |
| 1 | W1C | 0x0 | unlock_int Interrupt indicates that the controller received an Unlock message. 1'b0: No interrupt 1'b1: Interrupt |
| 0 | W1C | 0x0 | ven_msg_int Interrupt that indicates the controller received a vendor-defined message. The message header is available in VEN_MSG_RX_INFO registers. 1'b0: No interrupt 1'b1: Interrupt |

PCIE CLIENT INTR STATUS LEGACY

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:8 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 7 | RO | 0x0 | <p>tx_intd_int Emulation of sending the legacy PCI Interrupts. 1'b0 to 1'b1: The controller has send an Assert_INTD Message to the upstream device. 1'b1 to 1'b0: The controller has send an Deassert_INTD Message to the upstream device. EP mode only.</p> |
| 6 | RO | 0x0 | <p>tx_intc_int Emulation of sending the legacy PCI Interrupts. 1'b0 to 1'b1: The controller has send an Assert_INTC Message to the upstream device. 1'b1 to 1'b0: The controller has send an Deassert_INTC Message to the upstream device. EP mode only.</p> |
| 5 | RO | 0x0 | <p>tx_intb_int Emulation of sending the legacy PCI Interrupts. 1'b0 to 1'b1: The controller has send an Assert_INTB Message to the upstream device. 1'b1 to 1'b0: The controller has send an Deassert_INTB Message to the upstream device. EP mode only.</p> |
| 4 | RO | 0x0 | <p>tx_inta_int Emulation of sending the legacy PCI Interrupts. 1'b0 to 1'b1: The controller has send an Assert_INTA Message to the upstream device. 1'b1 to 1'b0: The controller has send an Deassert_INTA Message to the upstream device. EP mode only.</p> |
| 3 | RO | 0x0 | <p>rx_intd_int Emulation of reception of the legacy PCI Interrupts. RC mode only. 1'b0 to 1'b1: The controller received an Assert_INTD Message from the downstream device. 1'b1 to 1'b0: The controller received an Deassert_INTD Message from the downstream device. RC mode only.</p> |
| 2 | RO | 0x0 | <p>rx_intc_int Emulation of reception of the legacy PCI Interrupts. RC mode only. 1'b0 to 1'b1: The controller received an Assert_INTC Message from the downstream device. 1'b1 to 1'b0: The controller received an Deassert_INTC Message from the downstream device. RC mode only.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 1 | RO | 0x0 | <p>rx_intb_int Emulation of reception of the legacy PCI Interrupts. RC mode only.</p> <p>1'b0 to 1'b1: The controller received an Assert_INTB Message from the downstream device.</p> <p>1'b1 to 1'b0: The controller received an Deassert_INTB Message from the downstream device.</p> <p>RC mode only.</p> |
| 0 | RO | 0x0 | <p>rx_inta_int Emulation of reception of the legacy PCI Interrupts. RC mode only.</p> <p>1'b0 to 1'b1: The controller received an Assert_INTA Message from the downstream device.</p> <p>1'b1 to 1'b0: The controller received an Deassert_INTA Message from the downstream device.</p> <p>RC mode only.</p> |

PCIE CLIENT INTR STATUS ERR

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:13 | RO | 0x0 | reserved |
| 12 | W1C | 0x0 | <p>radm_qoverflow_int Interrupt indicates that one or more of the P/NP/CPL receive queues have overflowed.</p> <p>1'b0: No interrupt</p> <p>1'b1: Interrupt</p> |
| 11 | RO | 0x0 | reserved |
| 10 | W1C | 0x0 | <p>f_err_rx_int Interrupt indicates that the controller received an ERR_FATAL message.</p> <p>1'b0: No interrupt</p> <p>1'b1: Interrupt</p> <p>EP mode only.</p> |
| 9 | W1C | 0x0 | <p>nf_err_rx_int Interrupt indicates that the controller received an ERR_NONFATAL message.</p> <p>1'b0: no interrupt</p> <p>1'b1: interrupt</p> <p>EP mode only.</p> |
| 8 | W1C | 0x0 | <p>cor_err_rx_int Interrupt indicates that the controller received an ERR_COR message.</p> <p>1'b0: No interrupt</p> <p>1'b1: Interrupt</p> <p>EP mode only.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 7 | RO | 0x0 | reserved |
| 6 | W1C | 0x0 | f_err_sent_int EP has sent a message towards the Root Complex indicating that the EP has received a TLP that contained a fatal error. 1'b0: No interrupt 1'b1: Interrupt EP mode only. |
| 5 | W1C | 0x0 | nf_err_sent_int EP has sent a message towards the Root Complex indicating that the EP has received a TLP that contained a non-fatal error. 1'b0: No interrupt 1'b1: Interrupt |
| 4 | W1C | 0x0 | cor_err_sent_int EP has sent a message towards the Root Complex indicating that the EP has received a TLP that contained a correctable error. 1'b0: No interrupt 1'b1: Interrupt |
| 3 | W1C | 0x0 | tx_cpl_timeout_int Interrupt indicates that the application has not generated a completion for an incoming request within the required time interval. The default completion timeout value is approximately 10 ms. |
| 2 | W1C | 0x0 | rx_cpl_timeout_int Interrupt indicates that the completion TLP for a request has not been received within the expected time window. 1'b0: No interrupt 1'b1: Interrupt You can find the timed out completion information in RX_CPL_TIME_OUT_INFO register. |
| 1 | W1C | 0x0 | aer_rc_err_msi Interrupt asserted cfg_aer_rc_err_msi for one clock cycle when all of the following conditions are true: 1. MSI or MSI-X is enabled. 2. A reported error condition causes a bit to be set in the Root Error Status register. 3. The associated error message reporting enable bit is set in the Root Error Command register. The controller does not check if the associated MSI vector is unmasked. Note the difference between aer_rc_err_msi and aer_rc_err_int is that aer_rc_err_int is read only and rc_err_msi can be written 1 to clear. RC mode only. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RO | 0x0 | <p>aer_rc_err_int Interrupt asserted when a reported error condition causes a bit to be set in the Root Error Status register and the associated error message reporting enable bit is set in the Root Error Command register.</p> <p>aer_rc_err_int is set when the RC internally generates an error or when an error message is received by the RC.</p> <p>This signal is used when MSI/MSI-X is NOT enabled; otherwise see cfg_aer_rc_err_msi.</p> <p>RC mode only.</p> |

PCIE CLIENT INTR STATUS MISC

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:15 | RO | 0x0 | reserved |
| 14 | RO | 0x0 | <p>link_eq_req_int Interrupt indicating to your application that the Link Equalization Request bit in the Link Status 2 Register has been set and the Link Equalization Request Interrupt Enable (Link Control 3 Register bit 1) is set.</p> <p>1'b0: No interrupt 1'b1: Interrupt</p> <p>This bit is reserved for future use</p> |
| 13 | W1C | 0x0 | <p>rbar_update_int Interrupt indicates that a resizable BAR control register has been updated.</p> <p>1'b0: No interrupt 1'b1: Interrupt</p> |
| 12 | W1C | 0x0 | <p>dpa_sub_upd_int Interrupt asserted when the Substate Control field of the DPA Control Register has been updated with a new value. The controller asserts it when the updated value of the Substate Control field doesn't match the Substate Status field of the Status Register when the Substate Control Enable bit field of the Status Register is set to "1".</p> |
| 11 | RO | 0x0 | <p>edma_rd_int DMA read channel interrupt.</p> <p>Indicates that the DMA read transfer has completed or that an error has occurred.</p> <p>1'b0: No interrupt 1'b1: Interrupt</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 10 | RO | 0x0 | <p>edma_wr_int DMA write channel interrupt. Indicates that the DMA write transfer has completed or that an error has occurred.</p> <p>1'b0: No interrupt 1'b1: Interrupt</p> |
| 9 | W1C | 0x0 | <p>bw_mgt_msi Interrupt asserted when following conditions are true:</p> <ol style="list-style-type: none"> 1. MSI or MSI-X is enabled. 2. The Link Bandwidth Management Status register (Link Control Status register bit 14) is updated 3. The Link Bandwidth Management Interrupt Enable (Link Control register bit 10) is set. <p>RC mode only.</p> |
| 8 | RO | 0x0 | <p>bw_mgt_int Interrupt asserted when all of the following conditions are true:</p> <ol style="list-style-type: none"> 1. The INTx Assertion Disable bit in the Command register is 0 2. The Bandwidth Management Interrupt Enable bit in the Link Control register is set to 1 3. The Bandwidth Management Interrupt Status bit in the Link Status register is set to 1. <p>RC mode only.</p> |
| 7 | W1C | 0x0 | <p>link_auto_bw_msi The controller sets this pin when following conditions are true:</p> <ol style="list-style-type: none"> 1. MSI or MSI-X is enabled. 2. The Link Autonomous Bandwidth Status register (Link Status register bit 15) is updated. 3. The Link Autonomous Bandwidth Interrupt Enable (Link Control register bit 11) is set. <p>RC mode only.</p> |
| 6 | RO | 0x0 | <p>link_auto_bw_int Interrupt asserted when all of the following conditions are true:</p> <ol style="list-style-type: none"> 1. The INTx assertion disable bit in the Command register is 0, and 2. The Link Autonomous Bandwidth Interrupt Enable bit in the Link Control register is set to 1, and 3. The Link Autonomous Bandwidth Interrupt Status bit in the Link Status register is set to 1. <p>RC mode only.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | W1C | 0x0 | <p>hp_msi Interrupt asserted when following conditions are true:</p> <ol style="list-style-type: none"> 1. MSI or MSI-X is enabled. 2. Hot-Plug interrupts are enabled in the Slot Control register. 3. Any bit in the Slot Status register transitions from 0 to 1 and the associated event notification is enabled in the Slot Control register. <p>RC mode only.</p> |
| 4 | RO | 0x0 | <p>hp_int Interrupt asserted when all of the following conditions are true:</p> <ol style="list-style-type: none"> 1. The INTx Assertion Disable bit in the Command register is 0. 2. Hot-Plug interrupts are enabled in the Slot Control register. 3. Any bit in the Slot Status register is equal to 1, and the associated event notification is enabled in the Slot Control register. <p>hp_int stays asserted as long as the status bit is set. RC mode only.</p> |
| 3 | W1C | 0x0 | <p>hp_pme_int Interrupt asserted when all of the following conditions are true:</p> <ol style="list-style-type: none"> 1. The PME Enable bit in the Power Management Control and Status register is set to 1. 2. Any bit in the Slot Status register transitions from 0 to 1 and the associated event notification is enabled in the Slot Control register. <p>The controller does not check if the PM state is D1, D2, or D3hot. It is up to your application to check the value pm_dstate to make sure the device is in D1, D2, or D3hot. In addition, it asserts hp_pme only if PME is enabled, but it does not matter if hot-plug interrupts are enabled. RC mode only.</p> |
| 2 | W1C | 0x0 | <p>link_req_RST_not_int Interrupt for reset request because the link has gone down or the controller received a hot-reset request.</p> <p>1'b0: No interrupt 1'b1: Interrupt</p> |
| 1 | W1C | 0x0 | <p>dll_link_up_int Data link layer up/down status interrupt:</p> <p>1'b0: No interrupt 1'b1: Interrupt</p> |
| 0 | W1C | 0x0 | <p>phy_link_up_int PHY Link up/down status interrupt:</p> <p>1'b0: No interrupt 1'b1: Interrupt</p> |

PCIE CLIENT INTR STATUS PMC

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:9 | RO | 0x0 | reserved |
| 8 | W1 C | 0x0 | pm_dstate_update_int Interrupt indicates that the current power management D-state have changed. 1'b0: No interrupt 1'b1: Interrupt |
| 7 | W1 C | 0x0 | linkst_out_l0s_int Interrupt that indicates link has left L0s state(falling edge detected of pm_linkst_in_l0s in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |
| 6 | W1 C | 0x0 | linkst_out_l2_int Interrupt that indicates link has left L2 state(falling edge detected of pm_linkst_in_l2 in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |
| 5 | W1 C | 0x0 | linkst_out_l1_int Interrupt that indicates link has left L1 state(falling edge detected of pm_linkst_in_l1 in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |
| 4 | W1 C | 0x0 | linkst_out_l1sub_int Interrupt that indicates link has left L1 substate(falling edge detected of pm_linkst_in_l1sub in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |
| 3 | W1 C | 0x0 | linkst_in_l0s_int Interrupt that indicates link has entered L0s state(rising edge detected of pm_linkst_in_l0s in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |
| 2 | W1 C | 0x0 | linkst_in_l2_int Interrupt that indicates link has entered L2 state(rising edge detected of pm_linkst_in_l2 in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |
| 1 | W1 C | 0x0 | linkst_in_l1_int Interrupt that indicates link has entered L1 state(rising edge detected of pm_linkst_in_l1 in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | W1C | 0x0 | linkst_in_l1sub_int Interrupt that indicates link has entered L1 substate(rising edge detected of pm_linkst_in_l1sub in POWER_STATUS register). 1'b0: No interrupt 1'b1: Interrupt |

PCIE CLIENT INTR MASK MSG RX

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | WO | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15 | RO | 0x0 | reserved |
| 14 | RW | 0x1 | obff_cpu_active_int_mask Mask bit of the CPU Active OBFF message reception interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 13 | RW | 0x1 | obff_obff_int_mask Mask bit of the OBFF OBFF message reception interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 12 | RW | 0x1 | obff_idle_int_mask Mask bit of the OBFF IDLE message reception interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 11 | RO | 0x0 | reserved |
| 10 | RW | 0x1 | pm_turnoff_int_mask Mask bit of the pm_turnoff_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 9 | RW | 0x1 | pm_to_ack_int_mask Mask bit of the pm_to_ack_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 8 | RW | 0x1 | pm_pme_int_mask Mask bit of the pm_pme_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 7:6 | RO | 0x0 | reserved |
| 5 | RW | 0x1 | pme_msi_mask Mask bit of the pme_msi interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 4 | RW | 0x1 | pme_int_mask Mask bit of the pme_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 3 | RO | 0x0 | reserved |
| 2 | RW | 0x1 | ltr_msg_int_mask Mask bit of the LTR message reception interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 1 | RW | 0x1 | unlock_int_mask Mask bit of the unlock message reception interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 0 | RW | 0x1 | ven_msg_int_mask Mask bit of the vendor-defined message reception interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

PCIE CLIENT INTR MASK LEGACY

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | WO | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15:8 | RO | 0x0 | reserved |
| 7 | RW | 0x1 | tx_intd_int_mask Mask bit of the tx_intd_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 6 | RW | 0x1 | tx_intc_int_mask Mask bit of the tx_intc_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 5 | RW | 0x1 | tx_intb_int_mask Mask bit of the tx_intb_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 4 | RW | 0x1 | tx_inta_int_mask Mask bit of the tx_inta_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 3 | RW | 0x1 | rx_intd_int_mask Mask bit of the rx_intd_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 2 | RW | 0x1 | rx_intc_int_mask Mask bit of the rx_intc_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 1 | RW | 0x1 | rx_intb_int_mask Mask bit of the rx_intb_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 0 | RW | 0x1 | rx_inta_int_mask Mask bit of the rx_inta_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

PCIE CLIENT INTR MASK ERR

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | WO | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 15:13 | RO | 0x0 | reserved |
| 12 | RW | 0x1 | radm_qoverflow_mask Mask bit of the radmin_qoverflow_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 11 | RO | 0x0 | reserved |
| 10 | RW | 0x1 | f_err_rx_int_mask Mask bit of the f_err_rx_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 9 | RW | 0x1 | nf_err_rx_int_mask Mask bit of the nf_err_rx_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 8 | RW | 0x1 | cor_err_rx_int_mask Mask bit of the cor_err_rx_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 7 | RO | 0x0 | reserved |
| 6 | RW | 0x1 | f_err_sent_int_mask Mask bit of the f_err_sent_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 5 | RW | 0x1 | nf_err_sent_int_mask Mask bit of the nf_err_sent_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 4 | RW | 0x1 | cor_err_sent_int_mask Mask bit of the cor_err_sent_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 3 | RW | 0x1 | tx_cpl_timeout_int_mask Mask bit of the tx_cpl_timeout_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 2 | RW | 0x1 | rx_cpl_timeout_int_mask Mask bit of the rx_cpl_timeout_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 1 | RW | 0x1 | aer_rc_err_msi_mask Mask bit of the aer_rc_err_msi interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 0 | RW | 0x1 | aer_rc_err_int_mask Mask bit of the aer_rc_err_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

PCIE CLIENT INTR MASK MISC

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | WO | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15 | RO | 0x0 | reserved |
| 14 | RW | 0x1 | link_eq_req_int_mask Mask bit of the link_eq_req_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 13 | RW | 0x1 | rbar_updata_int_mask Mask bit of the rbar_updata_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 12 | RW | 0x1 | dpa_sub_upd_int_mask Mask bit of the dpa_sub_upd_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 11 | RW | 0x1 | edma_rd_int_mask Mask bit of the edma_rd_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 10 | RW | 0x1 | edma_wr_int_mask Mask bit of the edma_wr_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 9 | RW | 0x1 | bw_mgt_msi_mask Mask bit of the bw_mgt_msi interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 8 | RW | 0x1 | bw_mgt_int_mask Mask bit of the bw_mgt_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 7 | RW | 0x1 | link_auto_bw_msi_mask Mask bit of the link_auto_bw_msi interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 6 | RW | 0x1 | link_auto_bw_int_mask Mask bit of the link_auto_bw_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 5 | RW | 0x1 | hp_msi_mask Mask bit of the hp_msi interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 4 | RW | 0x1 | hp_int_mask Mask bit of the hp_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 3 | RW | 0x1 | hp_pme_int_mask Mask bit of the hp_pme_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 2 | RW | 0x1 | link_req_RST_not_int_mask Mask bit of the link_req_RST_not_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 1 | RW | 0x1 | dll_link_up_int_mask Mask bit of the dll_link_up_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 0 | RW | 0x1 | phy_link_up_int_mask Mask bit of the phy_link_up_int interrupt. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

PCIE CLIENT INTR MASK PMC

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15:9 | RO | 0x0 | reserved |
| 8 | RW | 0x1 | pm_dstate_update_int_mask Mask bit of the pm_dstate_update_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7 | RW | 0x1 | linkst_out_l0s_int_mask Mask bit of the linkst_out_l0s_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 6 | RW | 0x1 | linkst_out_l2_int_mask Mask bit of the linkst_out_l2_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 5 | RW | 0x1 | linkst_out_l1_int_mask Mask bit of the linkst_out_l1_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 4 | RW | 0x1 | linkst_out_l1sub_int_mask Mask bit of the linkst_out_l1sub_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 3 | RW | 0x1 | linkst_in_l0s_int_mask Mask bit of the linkst_in_l0s_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 2 | RW | 0x1 | linkst_in_l2_int_mask Mask bit of the linkst_in_l2_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 1 | RW | 0x1 | linkst_in_l1_int_mask Mask bit of the linkst_in_l1_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |
| 0 | RW | 0x1 | linkst_in_l1sub_int_mask Mask bit of the linkst_in_l1sub_int interrupt in INTR_STATUS_PMC register. 1'b0: Interrupt unmask 1'b1: Interrupt mask |

PCIE_CLIENT_POWER_CON

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | clk_req_n_con Valid when clk_req_n_bypass set to 1. 1'b0: pull down CLKREQ# 1'b1: tristate CLKREQ# |
| 12 | RW | 0x0 | clk_req_n_bypass 1'b0: clk_req_n not bypass, CLKREQ# driven by controller 1'b1: clk_req_n bypass, CLKREQ# driven by bit 13 of POWER_CON |
| 11:10 | RW | 0x0 | p2_cpm_disable 2'b00: when phy in P2 power state, enable clock PM for both L1 and L2. 2'b01: when phy in P2 power state, enable clock PM for L2 but disable for L1. 2'b10: when phy in P2 power state, disable clock PM for L2 but enable for L1. 2'b11: when phy in P2 power state, disable clock PM for both L1 and L2. |
| 9 | RW | 0x0 | app_clk_pm_en Clock PM feature enabled by application. Used to inhibit the programming of the Clock PM in Link Control Register. |
| 8 | RW | 0x0 | sys_aux_pwr_det Indicates if auxiliary power is present. 1'b0: Auxiliary power is not present 1'b1: Auxiliary power is present |
| 7:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | app_xfer_pending Indicates that your application has transfers pending. 1'b0: There are no transactions outside the controller. 1'b1: There are transactions outside the controller that the controller needs to transmit. For EP mode, prevents generation of requests to enter L1. Triggers exit if already in L1. For RC mode, triggers exit if already in L1. |
| 4 | RW | 0x0 | app_req_exit_l1 1'b0: Self clear to generate a pulse to controller. 1'b1: Application request to Exit L1. Request from your application to exit L1. It is only effective when L1 is enabled. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 3 | RW | 0x1 | <p>app_ready_entr_l23 1'b0: Application not ready to enter L23. 1'b1: Application ready to enter L23.</p> <p>Indication application that it is ready to enter the L23 state. The controller delays sending PM_Enter_L23 (in response to PM_Turn_Off) until this signal becomes active. When this signal has been asserted by the application, it must be kept asserted until L2 entry has completed.</p> <p>EP mode only.</p> |
| 2 | RW | 0x0 | <p>app_req_entr_l1 Application request to Enter L1 ASPM state. 1'b1: Application request to enter L1 state 1'b0: Self clear to generate a pulse to controller</p> <p>This bit is used by applications that need to control L1 entry instead of using the L1 entry timer as defined in the PCI Express Specification. It is only effective when L1 is enabled. The controller latches this request when in L0 or L0s; to be acted upon later.</p> <p>EP mode only.</p> |
| 1 | RW | 0x0 | <p>app_pm_xmt_pme Request controller to send PME message. Self-clear to generate a pulse to controller.</p> <p>If PME is enabled and PME support is configured for current PMCSR D-state asserting this signal will cause the controller to wake from either L1 or L2 state. When the controller has transitioned back to the L0 state it will transmit a PME message and set the PME_Status. Upon receiving the PME message the root complex should clear the PME_Status and change the D-state back to D0.</p> <p>This bit deasserted when D-state back to D0.</p> <p>EP mode only.</p> |
| 0 | RW | 0x0 | <p>app_clk_req_n Indicates that the application is ready to have reference clock removed. 1'b0: Application does not want to remove reference clock 1'b1: Application is ready to have reference clock removed</p> |

PCIE CLIENT POWER STATUS

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27 | RW | 0x0 | <p>pm_clkreq_in Status of wire CLKREQ#</p> |
| 26 | RW | 0x0 | <p>pm_clkreq_out Status of CLKREQ# drive signal</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 25 | RO | 0x0 | pm_wake_in Status of wire WAKE# |
| 24 | RO | 0x0 | pm_wake_out 1'b0: Controller is not in wake process or Link already back to L0 EP only. 1'b1: Controller assert WAKE# signal to request to make link back to L0 |
| 23 | RO | 0x0 | pm_linkst_in_l0s 1'b0: Power management is not in L0s. 1'b1: Power management is in L0s. |
| 22 | RO | 0x0 | pm_linkst_in_l2 1'b0: Power management is not in L2. 1'b1: Power management is in L2. |
| 21 | RO | 0x0 | pm_linkst_in_l1 1'b0: Power management is not in L1. 1'b1: Power management is in L1. |
| 20 | RO | 0x0 | pm_linkst_in_l1sub 1'b0: Power management is not in L1 substate. 1'b1: Power management is in L1 substate. |
| 19 | RO | 0x0 | reserved |
| 18:16 | RO | 0x0 | pm_l1sub_state Power management L1 sub-states FSM state. For debugging purposes, not for system operation. |
| 15:12 | RO | 0x0 | pm_slave_state Power management slave FSM state. For debugging purposes, not for system operation. |
| 11:9 | RO | 0x0 | reserved |
| 8:4 | RO | 0x00 | pm_master_state Power management master FSM state. For debugging purposes, not for system operation. |
| 3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | pm_curnt_state Indicates the current power state. For debugging purposes, not for system operation. |

PCIE CLIENT MSG GEN CON

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | WO | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 14 | RW | 0x0 | <p>obff_cpu_active_msg_req Request controller to generate a "CPU Active" OBFF message. This bit clears automatically when request has been sent. Only usable in a downstream port.</p> <p>When application sets OBFF Enable bit in Device Control 2 Register to choose to use OBFF message or WAKE# signaling.</p> |
| 13 | RW | 0x0 | <p>obff_obff_msg_req Request controller to generate a "OBFF" OBFF message. This bit clears automatically when request has been sent. Only usable in a downstream port.</p> <p>When application sets OBFF Enable bit in Device Control 2 Register to choose to use OBFF message or WAKE# signaling.</p> |
| 12 | RW | 0x0 | <p>obff_idle_msg_req Request controller to generate a "IDLE" OBFF message. This bit clears automatically when request has been sent. Only usable in a downstream port.</p> <p>When application sets OBFF Enable bit in Device Control 2 Register to choose to use OBFF message or WAKE# signaling.</p> |
| 11:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | <p>ltr_msg_req Set to request controller to send a LTR Message. This bit clears automatically when request is granted by controller. Application should put LTR message information ready in LTR_MSG_TX_INFO register before setting this bit.</p> |
| 7:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | <p>xmt_unlock_req Request controller to generate an Unlock message. This bit clears automatically when request has been sent.</p> |
| 4 | RW | 0x0 | <p>xmt_turnoff_req Request controller to generate a PME_Turn_Off message. This bit clears automatically when request has been sent.</p> |
| 3:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | <p>legacy_int_req This bit is intended to request controller to generate messages that emulates the legacy PCI Interrupts.</p> <p>When this bit goes from low to high, the controller generates an Assert_INTx Message. When this bit goes from high to low, the controller generates a Deassert_INTx Message.</p> <p>The Interrupt Pin register for the corresponding function determines which INTx Message the controller generates (INTA, INTB, INTC, or INTD).</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | RW | 0x0 | ven_msg_req Set to request controller to send a vendor-defined Message. This bit clears automatically when request is granted by controller. Application should put vendor message information ready in VEN_MSG_TX_CFGx registers before setting this bit. |

PCIE CLIENT MSI GEN CON

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | WO | 0x00000000 | msi_gen_req 32 bits write only MSI generation request signals. MSB has the highest priority and LSB has the lowest priority. Write 1 to a certain bit request the controller send a MSI interrupt. Software should use MSI capability register to access MSI mask or pending status. |

PCIE CLIENT MSI GEN FNUM TC

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:7 | RO | 0x0 | reserved |
| 6:4 | RW | 0x0 | msi_tc Traffic Class of the MSI request. Reserved for future use. |
| 3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x0 | msi_func_num The function number of the MSI request. Reserved for future use. |

PCIE CLIENT RBAR SIZE INFO0

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x00 | bar2_size_info BAR2 size information, same with BAR0 size field from the resizable BAR control register. Updated when rbar_updata_int interrupt occur. |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | bar1_size_info BAR1 size information, same with BAR0 size field from the resizable BAR control register. Updated when rbar_updata_int interrupt occur. |
| 7:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x00 | bar0_size_info BAR0 size information, same with BAR0 size field from the resizable BAR control register. Updated when rbar_updata_int interrupt occur. |

PCIE CLIENT RBAR SIZE INFO1

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x00 | bar5_size_info BAR5 size information, same with BAR0 size field from the resizable BAR control register. Updated when rbar_update_int interrupt occur. |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | bar4_size_info BAR4 size information, same with BAR0 size field from the resizable BAR control register. Updated when rbar_update_int interrupt occur. |
| 7:6 | RO | 0x0 | reserved |
| 5:0 | RW | 0x00 | bar3_size_info BAR3 size information, same with BAR0 size field from the resizable BAR control register. Updated when rbar_update_int interrupt occur. |

PCIE CLIENT DMA HSHAKE TOGG

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RO | 0x0 | reserved |
| 3 | RO | 0x0 | rdxfer_done_togg DMA read engine descriptor transfer done toggle signal. After a doorbell command, this signal is de-asserted. When a channel is operating in non-linked list mode this signal does not toggles |
| 2 | RO | 0x0 | wdxfer_done_togg DMA write engine descriptor transfer done toggle signal. After a doorbell command, this signal is de-asserted. When a channel is operating in non-linked list mode this signal does not toggles |
| 1 | RW | 0x0 | rdxfer_go_togg DMA read engine descriptor transfer go toggle signal. This signal is ignored whenever a DMA channel operates in non-linked mode. |
| 0 | RW | 0x0 | wdxfer_go_togg DMA Write engine descriptor transfer go toggle signal. This signal is ignored whenever a DMA channel operates in non-linked mode. |

PCIE CLIENT VEN MSG RX INFO0

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:16 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 15:0 | RO | 0x0000 | ven_msg_req_id The requester ID of the received Message. [15:8]: Bus number [7:3]: Device number [2:0]: Function number |

PCIE CLIENT VEN MSG RX INFO1

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | ven_msg_header_l The third double word of the Vendor Defined Message header. |

PCIE CLIENT VEN MSG RX INFO2

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RO | 0x00000000 | ven_msg_header_h The fourth double word of the Vendor Defined Message header. |

PCIE CLIENT VEN MSG TX CFG0

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x00 | ven_msg_code The Message Code for the vendor-defined Message TLP. |
| 23:16 | RW | 0x00 | ven_msg_tag Tag for the vendor-defined Message TLP. |
| 15:11 | RO | 0x0 | reserved |
| 10:9 | RW | 0x0 | ven_msg_attr The Attributes field for the vendor-defined Message TLP. bit1: Relaxed ordering. bit0: No snoop. |
| 8 | RW | 0x0 | ven_msg_ep The Poisoned TLP (EP) bit for the vendor-defined Message TLP. |
| 7:5 | RW | 0x0 | ven_msg_tc The Traffic Class field for the vendor-defined Message TLP. |
| 4:0 | RW | 0x00 | ven_msg_type The TYPE field for the vendor-defined Message TLP. |

PCIE CLIENT VEN MSG TX CFG1

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | ven_msg_data_l The third double word of the Vendor Defined Message header. |

PCIE CLIENT VEN MSG TX CFG2

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | ven_msg_data_h The fourth double word of the Vendor Defined Message header. |

PCIE CLIENT LTR MSG TX INFO

Address: Operational Base + offset (0x0068)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | ltr_msg_latency 32 bits latency information in LTR message that application wants to send. Please refer to PCI Express Specification for message format. |

PCIE CLIENT APP ERR RPT INFO0

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | err_hdr_log0 The header(bit 0 to bit 31) of the TLP that contained the error. |

PCIE CLIENT APP ERR RPT INFO1

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | err_hdr_log1 The header(bit 32 to bit 63) of the TLP that contained the error. |

PCIE CLIENT APP ERR RPT INFO2

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | err_hdr_log2 The header(bit 64 to bit 95) of the TLP that contained the error. |

PCIE CLIENT APP ERR RPT INFO3

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | err_hdr_log3 The header(bit 96 to bit 127) of the TLP that contained the error. |

PCIE CLIENT APP ERR RPT INFO4

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:17 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 16 | RW | 0x0 | err_advisory 1'b0: Application error is not an advisory error. 1'b1: Application error is an advisory error. |
| 15:13 | RO | 0x0 | reserved |
| 12:0 | RW | 0x0000 | err_bus The type of error that application detected. The controller combines the values err_bus bits with the internally-detected error signals to set the corresponding bit in the Uncorrectable or Correctable Error Status Registers, [0]: Malformed TLP [1]: Receiver Overflow [2]: Unexpected completion [3]: Completer abort [4]: Completion Timeout [5]: Unsupported request [6]: ECRC Check Failed [7]: Poisoned TLP received [8]: AtomicOp Egress Blocked [9]: Uncorrectable Internal Error [10]: Corrected Internal Error [11]: TLP Prefix Blocked Error Status [12]: ACS Violation |

PCIE CLIENT OBFF WAKE ELE CFG

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:24 | RW | 0xb4 | wk_min_f2f_wdt Configure the minimum falling edge to falling edge width. The minimum value Specification define is 700ns. Application set this value equals to wk_min_f2f_wdt * core_clock_period(8ns in gen1 and 4ns in gen2). |
| 23:16 | RW | 0xf0 | wk_max_f2f_wdt Configure the maximum falling edge to falling edge width. The maximum value Specification define is 1000ns. Application set this value equals to wk_max_f2f_wdt * core_clock_period(8ns in gen1 and 4ns in gen2). |
| 15:8 | RW | 0x50 | wk_mim_pls_wdt Configure the minimum WAKE# pulse width for both active-inactive-active and inactive-active-inactive pulse. The minumum value Specification define is 300ns. Application set this value equals to wk_mim_pls_wdt * core_clock_period(8ns in gen1 and 4ns in gen2). |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7:0 | RW | 0x64 | wk_max_pls_wdt Configure the maximum WAKE# pulse width for both active-inactive-active and inactive-active-inactive pulse. The maximum value Specification define is 500ns. Application set this value equals to wk_max_pls_wdt * core_clock_period(8ns in gen1 and 4ns in gen2). Setting this value less than 50% of wk_max_f2f_wdt * core_clock_period. |

PCIE CLIENT OBFF WAKE DEBUG

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | obff_wake_dec_err Error state of OBFF decoder FSM, used for debug. |
| 7:4 | RW | 0x0 | obff_wake_dec_state Current state of OBFF decoder FSM, used for debug. |
| 3 | RW | 0x0 | obff_wake_dec_cpu 1'b0: Do not initialize to CPU ACTIVE state 1'b1: Initialize the OBFF decoder to CPU ACTIVE state when obff_wake_dec_init is set |
| 2 | RW | 0x0 | obff_wake_dec_obff 1'b0: Do not initialize to OBFF state 1'b1: Initialize the OBFF decoder to OBFF state when obff_wake_dec_init is set |
| 1 | RW | 0x0 | obff_wake_dec_idle 1'b0: Do not initialize to IDLE state 1'b1: Initialize the OBFF decoder to IDLE state when obff_wake_dec_init is set |
| 0 | RW | 0x0 | obff_wake_dec_init 1'b0: Do not initialize the OBFF decoder 1'b1: Initialize the OBFF decoder |

PCIE CLIENT RX CPL TIME OUT INFO

Address: Operational Base + offset (0x0088)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | rx_timeout_cpl_tag The Tag field of the timed out completion. |
| 23:12 | RO | 0x000 | rx_timeout_cpl_len Length (in bytes) of the timed out completion. For a split completion, it indicates the number of bytes remaining to be delivered when the completion timed out. |
| 11:10 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 9:8 | RO | 0x0 | rx_timeout_cpl_attr The Attributes field of the timed out completion. |
| 7 | RO | 0x0 | reserved |
| 6:4 | RO | 0x0 | rx_timeout_cpl_tc The Traffic Class of the timed out completion. |
| 3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | rx_timeout_func_num The function Number of the timed out completion. |

PCIE CLIENT TX CPL TIME OUT INFO

Address: Operational Base + offset (0x008c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23:12 | RO | 0x000 | tx_timeout_cpl_len The Length of the timed out completion. |
| 11:10 | RO | 0x0 | reserved |
| 9:8 | RO | 0x0 | tx_timeout_cpl_attr The Attributes value of the timed out completion. |
| 7 | RO | 0x0 | reserved |
| 6:4 | RO | 0x0 | tx_timeout_cpl_tc The TC of the timed out completion. |
| 3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | tx_timeout_func_num The function number of the timed out completion. |

PCIE CLIENT TX CPL TIME OUT INFO2

Address: Operational Base + offset (0x0090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | trgt_lookup_empty indicates that the target completion LUT is not full |
| 15:8 | RW | 0x00 | trgt_lookup_id The target completion LUT lookup ID for the incoming request TLP. When using the optional target completion lookup table feature, the application must save the lookup ID and assert the same lookup ID client0/1/2_cpl_lookup_id when generating a completion for the request. |
| 7:0 | RW | 0x00 | trgt_timeout_lookup_id The target completion LUT lookup ID of the timed out completion |

PCIE CLIENT LOCAL CRU CTRL

Address: Operational Base + offset (0x0094)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | WO | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | radm_gate_disable Disable the radm_clk gating in local cru. |
| 12 | RW | 0x0 | aux_gate_disable Disable the aux_clk gating in local cru. |
| 11 | RW | 0x0 | link_down_gate_disable Disable the link down clock gating in local cru. |
| 10 | RW | 0x0 | dbi_aclk_gate_disable Disable the dbi_aclk gating in local cru. |
| 9 | RW | 0x0 | slv_aclk_gate_disable Disable the slv_aclk gating in local cru. |
| 8 | RW | 0x0 | mstr_aclk_gate_disable Disable the mstr_aclk gating in local cru. |
| 7:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | pcie_pm_phy_req_disable Mask the phy reset request from controller pm. |
| 2 | RW | 0x0 | pcie_pm_srst_req_disable Mask the sticky reset request from controller pm. |
| 1 | RW | 0x0 | pcie_pm_nsrst_req_disable Mask the non-sticky reset request from controller pm. |
| 0 | RW | 0x0 | pcie_pm_crst_req_disable Mask the core reset request from controller pm. |

PCIE CLIENT GENERAL DEBUG CON

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15:7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | sd_hold_ltssm 1'b0: Release LTSSM 1'b1: Hold LTSSM |
| 5 | RW | 0x0 | dbg_pba MSIX PBA RAM Debug Mode. Use this input to activate the debug mode and allow direct read/write access to the PBA. |
| 4 | RW | 0x0 | dbg_table MSIX Table RAM Debug Mode. Use this input to activate the debug mode and allow direct read/write access to the Table. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x0 | diag_ctrl_bus The rising edge of these two signals ([1:0]) enable the controller to assert an LCRC or ECRC to the packet that it currently being transferred: 3'bx01: Insert LCRC error by inverting the LSB of LCRC 3'bx10: Insert ECRC error by inverting the LSB of ECRC 3'b1xx: Select Fast Link Mode, simulation only |

PCIE CLIENT GENERAL DEBUG INFO

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | reserved |
| 15:10 | RO | 0x00 | smlh_ltssm_state, please refer to PCIe Appendix for state code. Current state of the LTSSM |
| 9 | RO | 0x0 | reserved |
| 8 | RO | 0x0 | radm_q_not_empty 1'b0: Receive queues do not contain TLP header/data. 1'b1: Receive queues contain TLP header/data |
| 7 | RO | 0x0 | radm_xfer_pending 1'b0: Receive TLP requests are not pending 1'b1: Receive TLP requests are pending |
| 6 | RO | 0x0 | edma_xfer_pending 1'b0: EDMA DBI transfer not pending 1'b1: EDMA transfer pending |
| 5 | RO | 0x0 | brdg_dbx_xfer_pending 1'b0: AXI Slave DBI transfer not pending 1'b1: AXI Slave DBI transfer pending |
| 4 | RO | 0x0 | brdg_slv_xfer_pending 1'b0: AXI Slave non-DBI transfer not pending 1'b1: AXI Slave non-DBI transfer pending |
| 3 | RO | 0x0 | reserved |
| 2 | RO | 0x0 | radm_idle 1'b0: RADM is not idle in status 1'b1: RADM is in idle status |
| 1 | RW | 0x0 | rdlh_link_up 1'b0: Data link layer down 1'b1: Data link layer up |
| 0 | RW | 0x0 | smlh_link_up 1'b0: PHY link is down 1'b1: PHY link is up |

PCIE CLIENT SLC DEBUG INFO COMMON

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | reserved |
| 30:28 | RW | 0x0 | <p>bus_select</p> <p>Select the 16 of total 78 bit of common silicon debug information:</p> <p>3'b000: Select bit 15 to 0 3'b001: Select bit 31 to 16 3'b010: Select bit 47 to 32 3'b011: Select bit 63 to 48 3'b100: Select bit 77 to 64 other: reserved</p> |
| 27:16 | RO | 0x0 | reserved |
| 15:0 | RO | 0x0000 | <p>sd_info_common</p> <p>16 of total 78 bit common silicon debug information, selected by bus_select field.</p> <p>[77:75]: l1sub_state: Level: L1 sub state [74]: init_eq_pending_g4: Level: Equalization sequence Gen4 [73]: init_eq_pending: Level: Equalization sequence Gen3 [72:61]: xdlh_curnt_seqnum[11:0]: Level: Tx TLP SEQ# [60:49]: rdlh_curnt_rx_ack_seqnum[11:0]: Level: Rx ACK SEQ# [48]: rdlh_vc0_initfc2_status: Level: Init-FC Flag2 VC0 [47]: rdlh_vc0_initfc1_status: Level: Init-FC Flag1 VC0 [46:45]: rdlh_dlcntrl_state[1:0]: Level: DLCM [44:37]: latched_ts_nfts[7:0]: Level: Latched NFTS [36:34]: ltssm_powerdown[1:0]: Level: PIPE: Power Down [33:18]: smlh_ltssm_variable[15:0]: Level: LTSSM Variable [17]: pm_pme_resend_flag: Pulse: PME Re-Send flag [16]: smlh_lane_reversed: Level: Lane Reversal Operation [15:9]: rmlh_framing_err_ptr[6:0]: Pulse: 1st Framing Error Pointer [8:5]: pm_slave_state[3:0]: Level: PM Internal State (Slave) [4:0]: pm_master_state[4:0]: Level: PM Internal State (Master)</p> |

PCIE CLIENT SLC DEBUG INFO_L0

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | reserved |
| 30:28 | RW | 0x0 | <p>bus_select</p> <p>Select the 16 of total 78 bit of IO silicon debug information:</p> <p>3'b000: Select bit 15 to 0 3'b001: Select bit 31 to 16 3'b010: Select bit 47 to 32 3'b011: Select bit 63 to 48 3'b100: Select bit 77 to 64 other: Reserved</p> |
| 27:16 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 15:0 | RO | 0x0000 | <p>sd_info_l0 16 of total 78 bit l0 silicon debug information, selected by bus_select field.</p> <p>[77:76]: eq_convergence_sts [1:0]: Level: Equalization convergence information Gen3</p> <p>[75]: eqpa_violate_rule_123[2]: Level: Rule C Violation Event Status Gen3</p> <p>[74]: eqpa_violate_rule_123[1]: Level: Rule B Violation Event Status Gen3</p> <p>[73]: eqpa_violate_rule_123[0]: Level: Rule A Violation Event Status Gen3</p> <p>[72]: mac_cdm_ras_des_reject_rtx: Level: Receive Reject Coefficient Event status Gen3</p> <p>[71:64]: phy_cdm_ras_des_fomfeedback: Level: Current Figure of Merit Gen3</p> <p>[63:61]: mac_cdm_ras_des_pset_lrx: Level: Current Local Receiver Preset Hint Gen3</p> <p>[60:55]: mac_cdm_ras_des_coef_ltx[5:0]: Level: Current Local Transmitter Pre Cursor coefficient Gen3</p> <p>[54:49]: mac_cdm_ras_des_coef_ltx[11:6]: Level: Current Local Transmitter Cursor coefficient Gen3</p> <p>[48:43]: mac_cdm_ras_des_coef_ltx[17:12]: Level: Current Local Transmitter Post Cursor coefficient Gen3</p> <p>[42:37]: mac_cdm_ras_des_coef_rtx[5:0]: Level: Current Remote Transmitter Pre Cursor coefficient Gen3</p> <p>[36:31]: mac_cdm_ras_des_coef_rtx[11:6]: Level: Current Remote Transmitter Cursor coefficient Gen3</p> <p>[30:25]: mac_cdm_ras_des_coef_rtx[17:12]: Level: Current Remote Transmitter Post Cursor coefficient Gen3</p> <p>[24:19]: mac_cdm_ras_des_if: Level: Remote Device LF Gen3</p> <p>[18:13]: mac_cdm_ras_des_fs: Level: Remote Device FS Gen3</p> <p>[12:5]: rmlh_deskew_fifo_ptr: Level: Deskew Pointer</p> <p>[4]: mac_phy_rxpolarity: Level: PIPE: RxPolarity</p> <p>[3]: latched_rxdetected: Level: PIPE: Detect Lane</p> <p>[2]: phy_mac_rxvalid_rxburst: Level: PIPE: RxValid/RxBurst</p> <p>[1]: phy_mac_rxelec_rxh8exit: Level: PIPE: RxElecIdle/RxHibern8ExitType1</p> <p>[0]: mac_phy_txelec_txburst: Level: PIPE: TxElecIdle/TxBurst</p> |

PCIE CLIENT SLC DEBUG INFO L1

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|-------------|
| 31 | RO | 0x0 | reserved |

| | | | |
|-------|----|-----|---|
| 30:28 | RW | 0x0 | bus_select Select the 16 of total 78 bit of l1 silicon debug information: 3'b000: Select bit 15 to 0 3'b001: Select bit 31 to 16 3'b010: Select bit 47 to 32 3'b011: Select bit 63 to 48 3'b100: Select bit 77 to 64 other: Reserved |
| 27:16 | RO | 0x0 | reserved |

| | | | |
|------|----|--------|--|
| | | | sd_info_l1 16 of total 78 bit l1 silicon debug information, selected by bus_select field. [77:76]: eq_convergence_sts [1:0]: Level: Equalization convergence information Gen3 [75]: eqpa_violate_rule_123[2]: Level: Rule C Violation Event Status Gen3 [74]: eqpa_violate_rule_123[1]: Level: Rule B Violation Event Status Gen3 [73]: eqpa_violate_rule_123[0]: Level: Rule A Violation Event Status Gen3 [72]: mac_cdm_ras_des_reject_rtx: Level: Receive Reject Coefficient Event status Gen3 [71:64]: phy_cdm_ras_des_fomfeedback: Level: Current Figure of Merit Gen3 [63:61]: mac_cdm_ras_des_pset_lrx: Level: Current Local Receiver Preset Hint Gen3 [60:55]: mac_cdm_ras_des_coef_ltx[5:0]: Level: Current Local Transmitter Pre Cursor coefficient Gen3 [54:49]: mac_cdm_ras_des_coef_ltx[11:6]: Level: Current Local Transmitter Cursor coefficient Gen3 [48:43]: mac_cdm_ras_des_coef_ltx[17:12]: Level: Current Local Transmitter Post Cursor coefficient Gen3 [42:37]: mac_cdm_ras_des_coef_rtx[5:0]: Level: Current Remote Transmitter Pre Cursor coefficient Gen3 [36:31]: mac_cdm_ras_des_coef_rtx[11:6]: Level: Current Remote Transmitter Cursor coefficient Gen3 [30:25]: mac_cdm_ras_des_coef_rtx[17:12]: Level: Current Remote Transmitter Post Cursor coefficient Gen3 [24:19]: mac_cdm_ras_des_if: Level: Remote Device LF Gen3 [18:13]: mac_cdm_ras_des_fs: Level: Remote Device FS Gen3 [12:5]: rmlh_deskew_fifo_ptr: Level: Deskew Pointer [4]: mac_phy_rxpolarity: Level: PIPE: RxPolarity [3]: latched_rxdetected: Level: PIPE: Detect Lane [2]: phy_mac_rxvalid_rxburst: Level: PIPE: RxValid/RxBurst [1]: phy_mac_rxelec_rxh8exit: Level: PIPE: RxElecIdle/RxHibern8ExitType1 [0]: mac_phy_txelec_txburst: Level: PIPE: TxElecIdle/TxBurst |
| 15:0 | RO | 0x0000 | PCIE CLIENT SLC DEBUG INFO VO Address: Operational Base + offset (0x0114) |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:28 | RW | 0x0 | <p>bus_select Select the 16 of total 240 bit of virtual channel 0 silicon debug information: 4'd0: Select bit 15 to 0 4'd1: Select bit 31 to 16 ... 4'd14: Select bit 239 to bit 224 other: Reserved</p> |
| 27:16 | RO | 0x0 | reserved |
| 15:0 | RO | 0x0000 | <p>sd_info_v0 16 of total 240 bit l1 silicon debug information, selected by bus_select field.</p> <p>[239:228]: rtlh_fc_allctd_cpld: Level: Credit Allocated (CD) [227:220]: rtlh_fc_allctd_cplh: Level: Credit Allocated (CH) [219:208]: rtlh_fc_allctd_npd: Level: Credit Allocated (ND) [207:200]: rtlh_fc_allctd_nph: Level: Credit Allocated (NH) [199:188]: rtlh_fc_allctd_pd: Level: Credit Allocated (PD) [187:180]: rtlh_fc_allctd_ph: Level: Credit Allocated (PH) [179:168]: rtlh_fc_rcvd_cpld: Level: Credit Received (CD) [167:160]: rtlh_fc_rcvd_cplh: Level: Credit Received (CH) [159:148]: rtlh_fc_rcvd_npd: Level: Credit Received (ND) [147:140]: rtlh_fc_rcvd_nph: Level: Credit Received (NH) [139:128]: rtlh_fc_rcvd_pd: Level: Credit Received (PD) [127:120]: rtlh_fc_rcvd_ph: Level: Credit Received (PH) [119:108]: xadm_fc_limit_cpld: Level: Credit Limit (CD) [107:100]: xadm_fc_limit_cplh: Level: Credit Limit (CH) [99:88]: xadm_fc_limit_npd: Level: Credit Limit (ND) [87:80]: xadm_fc_limit_nph: Level: Credit Limit (NH) [79:68]: xadm_fc_limit_pd: Level: Credit Limit (PD) [67:60]: xadm_fc_limit_ph: Level: Credit Limit (PH) [59:48]: xadm_fc_cnsmd_cpld: Level: Credit Consumed (CD) [47:40]: xadm_fc_cnsmd_cplh: Level: Credit Consumed (CH) [39:28]: xadm_fc_cnsmd_npd: Level: Credit Consumed (ND) [27:20]: xadm_fc_cnsmd_nph: Level: Credit Consumed (NH) [19:8]: xadm_fc_cnsmd_pd: Level: Credit Consumed (PD) [7:0]: xadm_fc_cnsmd_ph: Level: Credit Consumed (PH)</p> |

PCIE CLIENT DIAG STATUS BUS SEL

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:4 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 3:0 | RW | 0x0 | <p>diag_bus_sel Select the 32 of total 488 bit of diag_status_bus that present on register DIAG_STATUS_BUS_INFO.</p> <p>4'd0: bit 31 to 0 4'd1: bit 63 to 32 ... 4'd15: bit 487 to 480</p> |

PCIE CLIENT DIAG STATUS BUS INFO

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RO | 0x00000000 | <p>diag_status_bus 32 bit of total 488 bit width diag_status_bus. Selected by DIAG_STATUS_BUS_SEL register. Contains all of the important status signals from each controller module.</p> |

PCIE CLIENT AXI MSTR MISC CON

Address: Operational Base + offset (0x0200)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | WO | 0x0000 | <p>write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable</p> |
| 15:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | mstr_rmisc_ep EP bit of read cpl TLP |
| 3:2 | RW | 0x0 | <p>mstr_rmisc_cpl_status AXI Master Read Response selection bus. This bus controls the response sent on the PCIe wire in the case of successful read requests. 2'b00: SC (Successful Completion) 2'b01: CA (Completer Abort) 2'b10: UR (Unsupported Request) 2'b11: SC (Successful Completion)</p> |
| 1:0 | RW | 0x0 | <p>mstr_bmisc_cpl_status AXI Master Write Response selection bus. This controls the response to be sent on the wire in the case of successful write requests. 2'b00: SC (Successful Completion) 2'b01: CA (Completer Abort) 2'b10: UR (Unsupported Request) 2'b11: SC (Successful Completion)</p> |

PCIE CLIENT AXI SLV ATU BYPASS

Address: Operational Base + offset (0x0204)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable |
| 15:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | slv_ar_atu_bypass 1'b0: Not bypass 1'b1: AXI slave read address ATU bypass |
| 0 | RW | 0x0 | slv_aw_atu_bypass 1'b0: Not bypass 1'b1: AXI slave write address ATU bypass |

PCIE CLIENT AXI SLV AWMISC HDR

Address: Operational Base + offset (0x0208)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21:0 | RW | 0x0000000 | slv_awmisc_info [4:0]: TLP's TYPE [5]: Reserved [6]: TLP's EP bit [7]: Reserved [8]: TLP's NS bit [9]: TLP's RO bit [12:10]: TLP's TC bits [20:13]: TLP's MSG code [21]: AXI transaction is a DBI access. This is for SHARED DBI mode only. [24:22]: TLP's Function number. |

PCIE CLIENT AXI SLV AWMISC HDR3

Address: Operational Base + offset (0x020c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | slv_awmisc_info_hdr_3dw AXI Slave 3rd header DWs. The application drives this with the 3rd Header DWs it intends to send on a PCIe Msg/MsgD. |

PCIE CLIENT AXI SLV AWMISC HDR4

Address: Operational Base + offset (0x0210)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | slv_awmisc_info_hdr_4dw AXI Slave 4th header DWs. The application drives this with the 4th Header DWs it intends to send on a PCIe Msg/MsgD. |

PCIE CLIENT AXI SLV AWMISC TAG

Address: Operational Base + offset (0x0214)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | slv_awmisc_tag AXI Slave Write Request Tag. Sets the TAG number for output posted requests. It is expected that your application normally sets this to '0' except when generating ATS invalidate requests. |

PCIE CLIENT AXI SLV MISC INFO

Address: Operational Base + offset (0x0218)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:23 | RO | 0x0 | reserved |
| 22 | RW | 0x0 | slv_wmisc_info When asserted, the controller sets the Poisoned TLP (EP) bit in the TLP header of the current and subsequent Write Data transactions. |
| 21:0 | RW | 0x0000000 | slv_armisc_info Provide readTLP header information [4:0]: TLP's TYPE [5]: Reserved [6]: TLP's EP bit [7]: Reserved [8]: TLP's NS bit [9]: TLP's RO bit [12:10]: TLP's TC bits [20:13]: TLP's MSG code [21]: AXI transaction is a DBI access. This is for SHARED DBI mode only. [24:22]: TLP's function number. |

10.5.4 TYPE0 Header Registers Summary

PF PCI-Compatible Configuration Space Header Type0 Registers.

| Name | Offset | Size | Reset Value | Description |
|--|---------------|-------------|--------------------|---|
| TYPE0 DEVICE ID VEND OR ID REG | 0x0000 | W | 0x18081d87 | Device ID and Vendor ID Register |
| TYPE0 STATUS COMMAND REG | 0x0004 | W | 0x00100000 | Status and Command Register |
| TYPE0 CLASS CODE REVISION ID | 0x0008 | W | 0x00000001 | Class Code and Revision ID Register. |
| TYPE0 BIST HEADER TYPE LATENCY CACHE LINE SIZE REG | 0x000c | W | 0x00000000 | BIST, Header Type, Latency Timer, and Cache Line Size Register. |
| TYPE0 BAR0 REG | 0x0010 | W | 0x00000000 | BAR0 Register. |

| Name | Offset | Size | Reset Value | Description |
|--|--------|------|-------------|---|
| TYPE0_BAR1_REG | 0x0014 | W | 0x00000000 | BAR1 Register. |
| TYPE0_BAR2_REG | 0x0018 | W | 0x00000000 | BAR1 Register. |
| TYPE0_BAR3_REG | 0x001c | W | 0x00000000 | BAR3 Register. |
| TYPE0_BAR4_REG | 0x0020 | W | 0x00000000 | BAR4 Register. |
| TYPE0_BAR5_REG | 0x0024 | W | 0x00000000 | BAR5 Register. |
| TYPE0_CARDBUS_CIS_PT_R_REG | 0x0028 | W | 0x00000000 | CardBus CIS Pointer Register. |
| TYPE0_SUBSYSTEM_ID_SUBSYSTEM_VENDOR_ID_REG | 0x002c | W | 0x00000000 | Subsystem ID and Subsystem Vendor ID Register. |
| TYPE0_EXP_ROM_BASE_A_DDR_REG | 0x0030 | W | 0x00000000 | Expansion ROM BAR Register. |
| TYPE0_PCI_CAP_PTR_REG | 0x0034 | W | 0x00000040 | Capabilities Pointer Register. |
| TYPE0_MAX_LATENCY_MIN_GRANT_INTERRUPT_PIN_INTERRUPT_LINE_REG | 0x003c | W | 0x000001ff | Max_Lat, Min_Gnt, Interrupt Pin, and Interrupt Line Register. |

Notes: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.5 TYPE0 Header Detail Register Description

TYPE0 DEVICE ID VENDOR ID REG

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RO | 0x1808 | PCI_TYPE0_DEVICE_ID Device ID. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 15:0 | RO | 0x1d87 | PCI_TYPE0_VENDOR_ID Vendor ID. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 STATUS COMMAND REG

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 31 | RW | 0x0 | DETECTED_PARITY_ERR Detected Parity Error. This bit is set by a Function whenever it receives a Poisoned TLP, regardless of the state the Parity Error Response bit in the Command register. |
| 30 | RW | 0x0 | SIGNALLED_SYS_ERR Signaled System Error. This bit is set when a Function sends an ERR_FATAL or ERR_NONFATAL message, and the SERR# Enable bit in the Command register is 1b. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | RCVD_MASTER_ABORT Received Master Abort. This bit is set when a Requester receives a Completion with Unsupported Request Completion Status. |
| 28 | RW | 0x0 | RCVD_TARGET_ABORT Received Target Abort. This bit is set when a Requester receives a Completion with Completer Abort Completion Status. |
| 27 | RW | 0x0 | SIGNALLED_TARGET_ABORT Signaled Target Abort. This bit is set when a Function completes a Posted or Non-Posted Request as a Completer Abort error. |
| 26:25 | RO | 0x0 | DEV_SEL_TIMING DEVSEL Timing. This field was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this field to 00b. |
| 24 | RW | 0x0 | MASTER_DPE Master Data Parity Error. This bit is set by a Function if the Parity Error Response bit in the Command register is 1b and either of the following two conditions occurs: Function receives a Poisoned Completion Function transmits a Poisoned Request |
| 23 | RO | 0x0 | FAST_B2B_CAP Fast Back to Back Transaction Capable. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b. |
| 22 | RO | 0x0 | reserved |
| 21 | RO | 0x0 | FAST_66MHZ_CAP 66MHz Capable. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b. |
| 20 | RO | 0x1 | CAP_LIST Capabilities List. Indicates the presence of an Extended Capability list item. Since all PCI Express device Functions are required to implement the PCI Express Capability structure, the controller hardwires this bit to 1b. |
| 19 | RO | 0x0 | INT_STATUS Emulation interrupt pending. 1'b1: indicates that an INTx emulation interrupt is pending internally in the Function. Setting the Interrupt Disable bit has no effect on the state of this bit. |
| 18:11 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 10 | RW | 0x0 | <p>PCI_TYPE0_INT_EN Interrupt Disable. Controls the ability of a Function to generate INTx emulation interrupts.</p> <p>1'b1: Functions are prevented from asserting INTx interrupts. 1'b0: Functions are allowed to asserting INTx interrupts.</p> <p>Note: Any INTx emulation interrupts already asserted by the Function must be deasserted when this bit is Set. INTx interrupts use virtual wires that must, if asserted, be deasserted using the appropriate Deassert_INTx message(s) when this bit is set.</p> <p>Only the INTx virtual wire interrupt(s) associated with the Function(s) for which this bit is set are affected.</p> |
| 9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | <p>PCI_TYPE0_SERREN SERR# Enable. When set, this bit enables reporting upstream of Non-fatal and Fatal errors detected by the Function.</p> <p>Note: The errors are reported if enabled either through this bit or through the PCI Express specific bits in the Device Control register. For more details see the "Error Registers" section of the PCI Express Specification.</p> |
| 7 | RO | 0x0 | <p>PCI_TYPE_IDSEL_STEPPING IDSEL Stepping/Wait Cycle Control. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b.</p> |
| 6 | RW | 0x0 | <p>PCI_TYPE0_PARITY_ERR_EN Parity Error Response. This bit controls the logging of poisoned TLPs in the Master Data Parity Error bit in the Status register. For more details see the "Error Registers" section of the PCI Express Specification.</p> |
| 5 | RO | 0x0 | <p>PCI_TYPE_VGA_PALETTE_SNOOP VGA Palette Snoop. This bit was originally described in the PCI Local Bus Specification and the PCI-to-PCI Bridge architecture specification. Its functionality does not apply to PCI Express, the controller hardwires this bit to 0b.</p> |
| 4 | RO | 0x0 | <p>PCI_TYPE_MWI_ENABLE Memory Write and Invalidate. This bit was originally described in the PCI Local Bus Specification and the PCI-to-PCI Bridge architecture specification. Its functionality does not apply to PCI Express, the controller hardwires this bit to 0b.</p> |
| 3 | RO | 0x0 | <p>PCI_TYPE0_SPECIAL_CYCLE_OPERATION Special Cycle Enable. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 2 | RW | 0x0 | <p>PCI_TYPE0_BUS_MASTER_EN Bus Master Enable. Controls the ability of a Function to issue Memory and I/O Read/Write requests. 1'b1: the Function is allowed to issue Memory or I/O Requests. 1'b0: the Function is not allowed to issue any Memory or I/O Requests. Requests other than Memory or I/O Requests are not controlled by this bit.</p> |
| 1 | RW | 0x0 | <p>PCI_TYPE0_MEM_SPACE_EN Memory Space Enable. Controls a Function's response to Memory Space accesses. 1'b1: the Function is enabled to decode the address and further process Memory Space accesses. 1'b0: all received Memory Space accesses are caused to be handled as Unsupported Requests.</p> |
| 0 | RO | 0x0 | <p>PCI_TYPE0_IO_EN IO Space Enable. Controls a Function's response to I/O Space accesses. 1'b1: the Function is enabled to decode the address and further process I/O Space accesses. 1'b0: all received I/O accesses are caused to be handled as Unsupported Requests. For a Function that does not support I/O Space accesses, the controller hardwires this bit to 0b.</p> |

TYPE0 CLASS CODE REVISION ID

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:24 | RO | 0x00 | <p>BASE_CLASS_CODE Base Class Code. A code that broadly classifies the type of operation the Function performs. Encodings for base class, are provided in the PCI Code and ID Assignment Specification. All unspecified encodings are Reserved.</p> <p>Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |
| 23:16 | RO | 0x00 | <p>SUBCLASS_CODE Sub-Class Code. Specifies a base class sub-class, which identifies more specifically the operation of the Function. Encodings for sub-class are provided in the PCI Code and ID Assignment Specification. All unspecified encodings are Reserved.</p> <p>Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |
| 15:8 | RO | 0x00 | <p>PROGRAM_INTERFACE Programming Interface. This field identifies a specific register-level programming interface (if any) so that device independent software can interact with the Function. Encodings for interface are provided in the PCI Code and ID Assignment Specification. All unspecified encodings are Reserved.</p> <p>Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |
| 7:0 | RO | 0x01 | <p>REVISION_ID Revision ID. The value in this register specifies a Function specific revision identifier.</p> <p>Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |

TYPE0 BIST HEADER TYPE LATENCY CACHE LINE SIZE REG

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:24 | RO | 0x00 | <p>BIST BIST. This register is used for control and status of BIST. For Functions that do not support BIST the controller hardwires the register to 00h.</p> |
| 23 | RO | 0x0 | <p>MULTI_FUNC Multi-Function Device.</p> |
| 22:16 | RO | 0x00 | <p>HEADER_TYPE Header Layout. This field identifies the layout of the second part of the predefined header. The controller uses 000 0000b encoding.</p> |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 15:8 | RO | 0x00 | LATENCY_MASTER_TIMER Latency Timer. The Latency Timer was originally described in the PCI Local Bus Specification and the PCI-to-PCI Bridge Architecture Specification. Its functionality does not apply to PCI Express. The controller hardwires this register to 00h. |
| 7:0 | RW | 0x00 | CACHE_LINE_SIZE Cache Line Size. The Cache Line Size register is programmed by the system firmware or the operating system to system cache line size. However, legacy conventional PCI software may not always be able to program this register correctly especially in the case of Hot-Plug devices. This read-write register is implemented for legacy compatibility purposes but has no effect on any PCI Express device behavior. |

TYPE0 BAR0 REG

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:4 | RW | 0x00000000 | BAR0_START BAR0 Base Address. |
| 3 | RO | 0x0 | BAR0_PREFETCH If data is pre-fetchable: 1'b1: pre-fetchable 1'b0: not pre-fetchable Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 2:1 | RO | 0x0 | BAR0_TYPE Memory Space: Base Address registers that map into Memory Space can be 32 bits or 64 bits wide (to support mapping into a 64-bit address space). 2'b00: Base register is 32 bits wide and can be mapped anywhere in the 32 address bit Memory Space. 2'b01: Reserved. 2'b10: Base register is 64 bits wide and can be mapped anywhere in the 64 address bit Memory Space. 2'b11: Reserved. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 0 | RO | 0x0 | BAR0_MEM_IO BAR0 Memory Space Indicator. This bit is used to determine whether the register maps into Memory or I/O Space. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 BAR1 REG

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RW | 0x00000000 | BAR1_START BAR1 Base Address. |
| 3 | RO | 0x0 | BAR1_PREFETCH If data is pre-fetchable: 1'b1: pre-fetchable 1'b0: not pre-fetchable Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 2:1 | RO | 0x0 | BAR1_TYPE Memory Space: Base Address registers that map into Memory Space can be 32 bits or 64 bits wide (to support mapping into a 64-bit address space). 2'b00: Base register is 32 bits wide and can be mapped anywhere in the 32 address bit Memory Space. 2'b01: Reserved. 2'b10: Base register is 64 bits wide and can be mapped anywhere in the 64 address bit Memory Space. 2'b11: Reserved. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 0 | RO | 0x0 | BAR1_MEM_IO BAR1 Memory Space Indicator. This bit is used to determine whether the register maps into Memory or I/O Space. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 BAR2 REG

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RW | 0x00000000 | BAR2_START BAR2 Base Address. |
| 3 | RO | 0x0 | BAR2_PREFETCH If data is pre-fetchable: 1'b1: pre-fetchable 1'b0: not pre-fetchable Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 2:1 | RO | 0x0 | BAR2_TYPE Memory Space: Base Address registers that map into Memory Space can be 32 bits or 64 bits wide (to support mapping into a 64-bit address space). 2'b00: Base register is 32 bits wide and can be mapped anywhere in the 32 address bit Memory Space. 2'b01: Reserved. 2'b10: Base register is 64 bits wide and can be mapped anywhere in the 64 address bit Memory Space. 2'b11: Reserved. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 0 | RO | 0x0 | BAR2_MEM_IO BAR2 Memory Space Indicator. This bit is used to determine whether the register maps into Memory or I/O Space. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 BAR3 REG

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:4 | RW | 0x00000000 | BAR3_START BAR3 Base Address. |
| 3 | RO | 0x0 | BAR3_PREFETCH If data is pre-fetchable: 1'b1: pre-fetchable 1'b0: not pre-fetchable Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 2:1 | RO | 0x0 | <p>BAR3_TYPE Memory Space: Base Address registers that map into Memory Space can be 32 bits or 64 bits wide (to support mapping into a 64-bit address space).</p> <p>2'b00: Base register is 32 bits wide and can be mapped anywhere in the 32 address bit Memory Space.</p> <p>2'b01: Reserved.</p> <p>2'b10: Base register is 64 bits wide and can be mapped anywhere in the 64 address bit Memory Space.</p> <p>2'b11: Reserved.</p> <p>Wire: R (sticky)</p> <p>Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |
| 0 | RO | 0x0 | <p>BAR3_MEM_IO BAR3 Memory Space Indicator. This bit is used to determine whether the register maps into Memory or I/O Space.</p> <p>Wire: R (sticky)</p> <p>Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |

TYPE0 BAR4 REG

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:4 | RW | 0x00000000 | BAR4_START BAR4 Base Address. |
| 3 | RO | 0x0 | <p>BAR4_PREFETCH If data is pre-fetchable:</p> <p>1'b1: pre-fetchable</p> <p>1'b0: not pre-fetchable</p> <p>Wire: R (sticky)</p> <p>Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |
| 2:1 | RO | 0x0 | <p>BAR4_TYPE Memory Space: Base Address registers that map into Memory Space can be 32 bits or 64 bits wide (to support mapping into a 64-bit address space).</p> <p>2'b00: Base register is 32 bits wide and can be mapped anywhere in the 32 address bit Memory Space.</p> <p>2'b01: Reserved.</p> <p>2'b10: Base register is 64 bits wide and can be mapped anywhere in the 64 address bit Memory Space.</p> <p>2'b11: Reserved.</p> <p>Wire: R (sticky)</p> <p>Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RO | 0x0 | BAR4_MEM_IO BAR4 Memory Space Indicator. This bit is used to determine whether the register maps into Memory or I/O Space. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 BAR5 REG

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:4 | RW | 0x00000000 | BAR5_START BAR5 Base Address. |
| 3 | RO | 0x0 | BAR5_PREFETCH If data is pre-fetchable: 1'b1: pre-fetchable 1'b0: not pre-fetchable Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 2:1 | RO | 0x0 | BAR5_TYPE Memory Space: Base Address registers that map into Memory Space can be 32 bits or 64 bits wide (to support mapping into a 64-bit address space). 2'b00: Base register is 32 bits wide and can be mapped anywhere in the 32 address bit Memory Space. 2'b01: Reserved. 2'b10: Base register is 64 bits wide and can be mapped anywhere in the 64 address bit Memory Space. 2'b11: Reserved. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 0 | RO | 0x0 | BAR5_MEM_IO BAR5 Memory Space Indicator. This bit is used to determine whether the register maps into Memory or I/O Space. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 CARDBUS CIS PTR REG

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | CARDBUS_CIS_POINTER CardBus CIS Pointer. Its functionality does not apply to PCI Express. It is hardwired to 0000 0000h. |

TYPE0 SUBSYSTEM ID SUBSYSTEM VENDOR ID REG

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RO | 0x0000 | SUBSYS_DEV_ID Subsystem ID. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 15:0 | RO | 0x0000 | SUBSYS_VENDOR_ID Subsystem Vendor ID. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 EXP ROM BASE ADDR REG

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:11 | RO | 0x0000000 | EXP_ROM_BASE_ADDRESS Expansion ROM Base Address. Upper 21 bits of the Expansion ROM base address. The number of bits (out of these 21) that a Function actually implements depends on how much address space the Function requires. Wire: if (EXP_ROM_BAR_MASK_REG.ROM_BAR_ENABLED == 1) then R/W else R Dbi: if (EXP_ROM_BAR_MASK_REG.ROM_BAR_ENABLED == 1) then R/W else R |
| 10:1 | RO | 0x0 | reserved |
| 0 | RO | 0x0 | ROM_BAR_ENABLE Expansion ROM Enable. 1'b0: the Function's expansion ROM address space is disabled. 1'b1: address decoding is enabled using the parameters in the other part of the Expansion ROM Base Address register. Wire: if (EXP_ROM_BAR_MASK_REG.ROM_BAR_ENABLED == 1) then R/W else R Dbi: if (EXP_ROM_BAR_MASK_REG.ROM_BAR_ENABLED == 1) then R/W else R |

TYPE0 PCI CAP PTR REG

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RO | 0x40 | CAP_POINTER Capabilities Pointer. This register points to a valid capability structure. Either this structure is the PCI Express Capability structure, or a subsequent list item points to the PCI Express Capability structure. The bottom two bits are reserved, the controller sets it to 00b. Software must mask these bits off before using this register as a pointer in Configuration Space to the first entry of a linked list of new capabilities. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

TYPE0 MAX LATENCY MIN GRANT INTERRUPT PIN INTERRUPT LIN

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15:8 | RO | 0x01 | INT_PIN Interrupt Pin. The Interrupt Pin register identifies the legacy interrupt Message(s) the Function uses. The valid values are: 01h, 02h, 03h, and 04h: Map to legacy interrupt Messages for INTA, INTB, INTC, and INTD respectively. 00h: Indicates that the Function uses no legacy interrupt Message(s). 05h through FFh: Reserved. PCI Express defines one legacy interrupt Message for a single Function device and up to four legacy interrupt Messages for a multi-Function device. For a single Function device, only INTA may be used. Any Function on a multi-Function device can use any of the INTx Messages. If a device implements a single legacy interrupt Message, it must be INTA; if it implements two legacy interrupt Messages, they must be INTA and INTB; and so forth. For a multi-Function device, all Functions may use the same INTx Message or each may have its own (up to a maximum of four Functions) or any combination thereof. A single Function can never generate an interrupt request on more than one INTx Message. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 7:0 | RW | 0xff | <p>INT_LINE</p> <p>Interrupt Line. The Interrupt Line register communicates interrupt line routing information. The register must be implemented by any Function that uses an interrupt pin. Values in this register are programmed by system software and are system architecture specific. The Function itself does not use this value; rather the value in this register is used by device drivers and operating systems.</p> |

10.5.6 TYPE1 Header Registers Summary

PF PCI-Compatible Configuration Space Header Type1 Registers.

| Name | Offset | Size | Reset Value | Description |
|--|--------|------|-------------|--|
| TYPE1_TYPE1_DEV_ID_VENDOR_ID_REG | 0x0000 | W | 0x18081d87 | Device ID and Vendor ID Register. |
| TYPE1_TYPE1_STATUS_COMMAND_REG | 0x0004 | W | 0x00100000 | Status and Command Register. |
| TYPE1_TYPE1_CLASS_CODE_REVISION_ID_REG | 0x0008 | W | 0x00000001 | Class Code and Revision ID Register. |
| TYPE1_TYPE1_BIST_HDR | | | | BIST, Header Type, Latency |
| TYPE1_TYPE1_CACHE_LINE_SIZE_REG | 0x000c | W | 0x00010000 | Timer, and Cache Line Size Register. |
| TYPE1_SEC_LAT_TIMER | | | | Secondary Latency Timer, |
| SUB_BUS_SEC_BUS_PRIMARY_BUS_REG | 0x0018 | W | 0x00000000 | Subordinate Bus Number, Secondary Bus Number, and Primary Bus Number Register. |
| TYPE1_SEC_STATUS_I_O_LIMIT | 0x001c | W | 0x00000000 | Secondary Status, and I/O Limit and Base Register. |
| TYPE1_MEMORY_LIMIT | | | | Memory Limit and Base Register. |
| TYPE1_PREF_MEMORY_LIMIT | 0x0020 | W | 0x00000000 | Prefetchable Memory Limit and Base Register. |
| TYPE1_PREF_MEMORY_BASE | 0x0024 | W | 0x00010001 | Prefetchable Memory Limit and Base Register. |
| TYPE1_PREF_BASE_UPPER | 0x0028 | W | 0x00000000 | Prefetchable Base Upper 32 Bits Register. |
| TYPE1_PREF_LIMIT_UPPER | 0x002c | W | 0x00000000 | Prefetchable Limit Upper 32 Bits Register. |
| TYPE1_I_O_LIMIT_UPPER | 0x0030 | W | 0x00000000 | I/O Limit and Base Upper 16 Bits Register. |
| TYPE1_TYPE1_CAPABILITY_POINTER | 0x0034 | W | 0x00000040 | Capabilities Pointer Register. |
| TYPE1_EXPANSION_ROM_ADDRESS | 0x0038 | W | 0x00000000 | Expansion ROM Base Address Register. |
| TYPE1_BRIDGE_CTRL_INTERRUPT | | | | Bridge Control, Interrupt Pin, and |
| PIN_INTERRUPT_LINE_REG | 0x003c | W | 0x000001ff | Interrupt Line Register. |

10.5.7 TYPE1 Header Detail Register Description**TYPE1 TYPE1 DEV ID VEND ID REG**

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x1808 | DEVICE_ID Device ID. The Device ID register identifies the particular Function. This identifier is allocated by the vendor. Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 15:0 | RO | 0x1d87 | VENDOR_ID Vendor ID. The Vendor ID register identifies the manufacturer of the Function. Valid vendor identifiers are allocated by the PCI-SIG to ensure uniqueness. It is not permitted to populate this register with a value of FFFFh, which is an invalid value for Vendor ID. Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

Notes: **S**-Size: **B**- Byte (8 bits) access, **H**W- Half WORD (16 bits) access, **W**-WORD (32 bits) access**TYPE1 TYPE1 STATUS COMMAND REG**

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | DETECTED_PARITY_ERROR Detected Parity Error. This bit is set by a Function whenever it receives a Poisoned TLP, regardless of the state the Parity Error Response bit in the Command register. The bit is set when the Poisoned TLP is received by a Function's primary side. |
| 30 | RW | 0x0 | SIGNALLED_SYS_ERROR Signaled System Error. This bit is set when a Function sends an ERR_FATAL or ERR_NONFATAL Message, and the SERR# Enable bit in the Command register is 1b. |
| 29 | RW | 0x0 | RCVD_MASTER_ABORT Received Master Abort. This bit is set when a Requester receives a Completion with Unsupported Request Completion status. The bit is set when the Unsupported Request is received by a Function's primary side. |
| 28 | RW | 0x0 | RCVD_TARGET_ABORT Received Target Abort. This bit is set when a Requester receives a Completion with Completer Abort Completion status. The bit is set when the Completer Abort is received by a Function's primary side. |
| 27 | RW | 0x0 | SIGNALLED_TARGET_ABORT Signaled Target Abort. This bit is set when a Function completes a Posted or Non-Posted Request as a Completer Abort error. This applies to a Function when the Completer Abort was generated by its primary side. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 26:25 | RO | 0x0 | DEV_SEL_TIMING DEVSEL Timing. This field was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires it to 00b. |
| 24 | RW | 0x0 | MASTER_DPE Master Data Parity Error. This bit is set by a Function if the Parity Error Response bit in the Command register is 1b and either of the following two conditions occurs: Port receives a Poisoned Completion going downstream Port transmits a Poisoned Request upstream If the Parity Error Response bit is 0b, this bit is never set. |
| 23 | RO | 0x0 | FAST_B2B_CAP Fast Back-to-Back Transactions Capable. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b. |
| 22 | RO | 0x0 | reserved |
| 21 | RO | 0x0 | FAST_66MHZ_CAP 66 MHz Capable. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b. |
| 20 | RO | 0x1 | CAP_LIST Capabilities List. Indicates the presence of an Extended Capability list item. Since all PCI Express device Functions are required to implement the PCI Express Capability structure, the controller hardwires this bit to 1b. |
| 19 | RO | 0x0 | INT_STATUS Interrupt Status. When set, indicates that an INTx emulation interrupt is pending internally in the Function. INTx emulation interrupts forwarded by Functions from the secondary side are not reflected in this bit. Setting the Interrupt Disable bit has no effect on the state of this bit. For Functions that do not generate INTx interrupts, the controller hardwires this bit to 0b. |
| 18:11 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 10 | RW | 0x0 | <p>INT_EN</p> <p>Interrupt Disable. Controls the ability of a Function to generate INTx emulation interrupts. When set, Functions are prevented from asserting INTx interrupts.</p> <p>Note:</p> <p>Any INTx emulation interrupts already asserted by the Function must be deasserted when this bit is set. INTx interrupts use virtual wires that must, if asserted, be deasserted using the appropriate Deassert_INTx message(s) when this bit is set. Only the INTx virtual wire interrupt(s) associated with the Function(s) for which this bit is set are affected.</p> <p>For Functions that generate INTx interrupts on their own behalf, this bit is required. This bit has no effect on interrupts forwarded from the secondary side. For Functions that do not generate INTx interrupts on their own behalf this bit is optional. If this bit is not implemented, the controller hardwires it to 0b.</p> |
| 9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | <p>SERREN</p> <p>SERR# Enable. When set, this bit enables reporting upstream of Non-fatal and Fatal errors detected by the Function.</p> <p>Note: The errors are reported if enabled either through this bit or through the PCI Express specific bits in the Device Control register. For more details see the "Error Registers" section of the PCI Express Specification.</p> <p>In addition, this bit controls transmission by the primary interface of ERR_NONFATAL and ERR_FATAL error Messages forwarded from the secondary interface. This bit does not affect the transmission of forwarded ERR_COR messages.</p> |
| 7 | RO | 0x0 | <p>IDSEL</p> <p>IDSEL Stepping/Wait Cycle Control. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b.</p> |
| 6 | RW | 0x0 | <p>PERREN</p> <p>Parity Error Response. This bit controls the logging of poisoned TLPs in the Master Data Parity Error bit in the Status register. For more details see the "Error Registers" section of the PCI Express Specification.</p> |
| 5 | RO | 0x0 | <p>VGAPS</p> <p>VGA Palette Snoop. This bit was originally described in the PCI Local Bus Specification and the PCI-to-PCI Bridge Architecture Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 4 | RO | 0x0 | <p>MWI_EN Memory Write and Invalidate. This bit was originally described in the PCI Local Bus Specification and the PCI-to-PCI Bridge Architecture Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b. For PCI Express to PCI/PCI-X Bridges, refer to the PCI Express to PCI/PCI-X Bridge Specification for requirements for this register.</p> |
| 3 | RO | 0x0 | <p>SCO Special Cycle Enable. This bit was originally described in the PCI Local Bus Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b.</p> |
| 2 | RW | 0x0 | <p>BME Bus Master Enable. This bit controls forwarding of Memory or I/O requests by a port in the Upstream direction. When this bit is 0b, Memory and I/O Requests received at a Root Port must be handled as Unsupported Requests (UR) For Non-Posted Requests a Completion with UR completion status must be returned. This bit does not affect forwarding of Completions in either the Upstream or Downstream direction. The forwarding of Requests other than Memory or I/O Requests is not controlled by this bit.</p> |
| 1 | RW | 0x0 | <p>MSE Memory Space Enable. This bit controls a Function's response to Memory Space accesses received on its primary side. When set, the Function is enabled to decode the address and further process Memory Space accesses. When clear, all received Memory Space accesses are caused to be handled as Unsupported Requests. You cannot write to this register if your configuration has no MEM bars; that is, the internal signal has_mem_bar =0. Note: The access attributes of this field are as follows: Wire: No access. Dbi: !has_mem_bar ? RO : RW</p> |
| 0 | RW | 0x0 | <p>IO_EN IO Space Enable. This bit controls a Function's response to I/O Space accesses received on its primary side. When set, the Function is enabled to decode the address and further process I/O Space accesses. When clear, all received I/O accesses are caused to be handled as Unsupported Requests. You cannot write to this register if your configuration has no IO bars; that is, the internal signal has_io_bar =0. Note: The access attributes of this field are as follows: Wire: No access. Dbi: !has_io_bar ? RO : RW</p> |

TYPE1 TYPE1 CLASS CODE REV ID REG

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x00 | <p>BASE_CLASS_CODE Base Class Code. A code that broadly classifies the type of operation the Function performs. Encodings for base class, are provided in the PCI Code and ID Assignment Specification. All unspecified encodings are reserved.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky.</p> |
| 23:16 | RO | 0x00 | <p>SUBCLASS_CODE Sub-Class Code. Specifies a base class sub-class, which identifies more specifically the operation of the Function. Encodings for sub-class are provided in the PCI Code and ID Assignment Specification. All unspecified encodings are reserved.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky.</p> |
| 15:8 | RO | 0x00 | <p>PROGRAM_INTERFACE Programming Interface. This field identifies a specific register level programming interface (if any) so that device independent software can interact with the Function. Encodings for interface are provided in the PCI Code and ID Assignment Specification. All unspecified encodings are reserved.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky.</p> |
| 7:0 | RO | 0x01 | <p>REVISION_ID Revision ID. The value of this field specifies a Function specific revision identifier. The value is chosen by the vendor. Zero is an acceptable value. The Revision ID should be viewed as a vendor defined extension to the Device ID.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky.</p> |

TYPE1 TYPE1 BIST HDR TYPE LAT CACHE LINE SIZE REG

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:24 | RO | 0x00 | <p>BIST BIST. This register is used for control and status of BIST. Functions that do not support BIST must hardwire the register to 00h. A Function whose BIST is invoked must not prevent normal operation of the PCI Express Link.</p> <p>Bit descriptions:</p> <ul style="list-style-type: none"> [31]: BIST Capable. When set, this bit indicates that the Function supports BIST. When Clear, the Function does not support BIST. [30]: Start BIST. If BIST Capable is set, set this bit to invoke BIST. The Function resets the bit when BIST is complete. Software is permitted to fail the device if this bit is not Clear (BIST is not complete) 2 seconds after it had been set. Writing this bit to 0b has no effect. The controller hardwires this bit to 0b if BIST Capable is clear. [29:28]: Reserved. [27:24]: Completion Code. This field encodes the status of the most recent test. A value of 0000b means that the Function has passed its test. Non-zero values mean the Function failed. Function-specific failure codes can be encoded in the non-zero values. This field's value is only meaningful when BIST Capable is set and Start BIST is Clear. This field must be hardwired to 0000b if BIST Capable is clear. |
| 23 | RO | 0x0 | <p>MULTI_FUNC Multi-Function Device. When set, indicates that the device may contain multiple Functions, but not necessarily. Software is permitted to probe for Functions other than Function 0. When clear, software must not probe for Functions other than Function 0 unless explicitly indicated by another mechanism, such as an ARI or SR-IOV Capability structure. Except where stated otherwise, it is recommended that this bit be set if there are multiple Functions, and clear if there is only one Function.</p> <p>Note: This register field is sticky.</p> |
| 22:16 | RO | 0x01 | <p>HEADER_TYPE Header Layout. This field identifies the layout of the second part of the predefined header. The controller uses 000 0001b encoding. The encoding 000 0010b is reserved. This encoding was originally described in the PC Card Standard Electrical Specification and is used in previous versions of the programming model. Careful consideration should be given to any attempt to repurpose it.</p> |

| | | | |
|------|----|------|--|
| 15:8 | RO | 0x00 | LATENCY_MASTER_TIMER Latency Timer. This register is also referred to as Primary Latency Timer. The Latency Timer was originally described in the PCI Local Bus Specification and the PCI-to-PCI Bridge Architecture Specification. Its functionality does not apply to PCI Express. The controller hardwires this register to 00h. |
| 7:0 | RW | 0x00 | CACHE_LINE_SIZE Cache Line Size. The Cache Line Size register is programmed by the system firmware or the operating system to system cache line size. However, legacy conventional PCI software may not always be able to program this register correctly especially in the case of Hot-Plug devices. This read-write register is implemented for legacy compatibility purposes but has no effect on any PCI Express device behavior. |

TYPE1 SEC LAT TIMER SUB BUS SEC BUS PRI BUS REG

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:24 | RO | 0x00 | SEC_LAT_TIMER Secondary Latency Timer. This register does not apply to PCI Express. The controller hardwires it to 00h. |
| 23:16 | RW | 0x00 | SUB_BUS Subordinate Bus Number. The Subordinate Bus Number register is used to record the bus number of the highest numbered PCI bus segment which is behind (or subordinate to) the bridge. Configuration software programs the value in this register. The bridge uses this register in conjunction with the Secondary Bus Number register to determine when to respond to and pass on a Type 1 configuration transaction on the primary interface to the secondary interface. |
| 15:8 | RW | 0x00 | SEC_BUS Secondary Bus Number. The Secondary Bus Number register is used to record the bus number of the PCI bus segment to which the secondary interface of the bridge is connected. Configuration software programs the value in this register. The bridge uses this register to determine when to respond to and convert a Type 1 configuration transaction on the primary interface into a Type 0 transaction on the secondary interface. |
| 7:0 | RW | 0x00 | PRIM_BUS Primary Bus Number. This register is not used by PCI Express Functions. It is implemented for compatibility with legacy software. |

TYPE1 SEC STAT IO LIMIT IO BASE REG

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | SEC_STAT_DPE Detected Parity Error. This bit is set by a Function when a Poisoned TLP is received by its secondary side, regardless of the state the Parity Error Response Enable bit in the Bridge Control register. |
| 30 | RW | 0x0 | SEC_STAT_RCVD_SYS_ERR Received System Error. This bit is set when the secondary side of a Function receives an ERR_FATAL or ERR_NONFATAL message. |
| 29 | RW | 0x0 | SEC_STAT_RCVD_MSTR_ABRT Received Master Abort. This bit is set when the secondary side of a Function (for requests initiated by the Type 1 header Function itself) receives a Completion with Unsupported Request Completion status. |
| 28 | RW | 0x0 | SEC_STAT_RCVD_TRGT_ABRT Received Target Abort. This bit is set when the secondary side of a Function (for requests initiated by the Type 1 header Function itself) receives a Completion with Completer Abort Completion status. |
| 27 | RW | 0x0 | SEC_STAT_SIG_TRGT_ABRT Signaled Target Abort. This bit is set when the secondary side of the Function (for Requests completed by the Type 1 header Function itself) completes a Posted or Non-Posted request as a Completer Abort error. |
| 26:25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | SEC_STAT_MDPE Master Data Parity Error. This bit is set by a Function if the Parity Error Response Enable bit in the Bridge Control register is set, and either of the following two conditions occurs: Port receives a Poisoned Completion coming Upstream Port transmits a Poisoned Request Downstream If the Parity Error Response Enable bit is clear, this bit is never set. |
| 23:16 | RO | 0x0 | reserved |
| 15:12 | RW | 0x0 | IO_LIMIT I/O Limit Address. These bits correspond to the address[15:12] of IO address range. For the purpose of address decoding, the bridge assumes that the lower 12 address bits, address[11:0], of the I/O limit address (not implemented in the I/O Limit register) are FFFh. The I/O Limit register can be programmed to a smaller value than the I/O Base register, if there are no I/O addresses on the secondary side of the bridge. In this case, the bridge will not forward any I/O transactions from the primary bus to the secondary and will forward all I/O transactions from the secondary bus to the primary bus. |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 11:9 | RO | 0x0 | reserved |
| 8 | RO | 0x0 | <p>IO_DECODE_BIT8 I/O Addressing Encode (IO Limit Address). This bit encodes the I/O addressing capability of the bridge. IO_DECODE_BIT8 indicates the following:</p> <p>0h: The bridge supports only 16-bit I/O addressing (for ISA compatibility). For the purpose of address decoding, the bridge assumes that the upper 16 address bits, Address[31:16], of the I/O limit address (not implemented in I/O Limit register) are zero.</p> <p>Note: The bridge must still perform a full 32-bit decode of the I/O address (that is, check that Address[31:16] are 0000h). In this case, the I/O address range supported by the bridge will be restricted to the first 64 KB of I/O Space (0000 0000h to 0000 FFFFh).</p> <p>01h: The bridge supports 32-bit I/O address decoding, and the I/O Limit Upper 16 Bits hold the upper 16 bits, corresponding to Address[31:16], of the 32-bit Limit address. In this case, system configuration software is permitted to locate the I/O address range supported by the bridge anywhere in the 4-GB I/O Space.</p> <p>Note: The 4-KB alignment and granularity restrictions still apply when the bridge supports 32-bit I/O addressing.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: No access.</p> <p>Dbi: R</p> |
| 7:4 | RW | 0x0 | <p>IO_BASE I/O Base Address. These bits correspond to the address[15:12] of I/O address range. For the purpose of address decoding, the bridge assumes that the lower 12 address bits, address[11:0], of the I/O base address (not implemented in the I/O Base register) are zero.</p> <p>Value After Reset: 0x0</p> |
| 3:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | RO | 0x0 | <p>IO_DECODE I/O Addressing Encode (IO Base Address) This bit encodes the IO addressing capability of the bridge. IO_DECODE indicates the following:</p> <p>0h: The bridge supports only 16-bit I/O addressing (for ISA compatibility). For the purpose of address decoding, the bridge assumes that the upper 16 address bits, Address[31:16], of the I/O base address (not implemented in I/O base register) are zero.</p> <p>Note: The bridge must still perform a full 32-bit decode of the I/O address (that is, check that Address[31:16] are 0000h). In this case, the I/O address range supported by the bridge will be restricted to the first 64 KB of I/O Space (0000 0000h to 0000 FFFFh).</p> <p>01h: The bridge supports 32-bit I/O address decoding, and the I/O Base Upper 16 Bits hold the upper 16 bits, corresponding to Address[31:16], of the 32-bit Base address. In this case, system configuration software is permitted to locate the I/O address range supported by the bridge anywhere in the 4-GB I/O Space.</p> <p>Note: The 4-KB alignment and granularity restrictions still apply when the bridge supports 32-bit I/O addressing.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: No access.</p> <p>Dbi: if (DBI_RO_WR_EN == 1) then R/W else R</p> |

TYPE1 MEM LIMIT MEM BASE REG

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:20 | RW | 0x000 | <p>MEM_LIMIT Memory Limit Address. These bits correspond to the upper 12 address bits, Address[31:20], of 32-bit addresses. For the purpose of address decoding, the bridge assumes that the lower 20 address bits, Address[19:0], of the memory limit address (not implemented in the Memory Limit register) are F FFFFh. The Memory Limit register must be programmed to a smaller value than the Memory Base register if there is no memory-mapped address space on the secondary side of the bridge.</p> |
| 19:16 | RO | 0x0 | reserved |
| 15:4 | RW | 0x000 | <p>MEM_BASE Memory Base Address. These bits correspond to the upper 12 address bits, Address[31:20], of 32-bit addresses. For the purpose of address decoding, the bridge assumes that the lower 20 address bits, Address[19:0], of the memory base address (not implemented in the Memory Base register) are zero.</p> |
| 3:0 | RO | 0x0 | reserved |

TYPE1 PREF MEM LIMIT PREF MEM BASE REG

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RW | 0x000 | PREF_MEM_LIMIT Prefetchable Memory Limit Address. If the Prefetchable Memory Limit register indicates support for 32-bit addressing, then the Prefetchable Limit Upper 32 Bits register is implemented as a read-only register that returns zero when read. If the Prefetchable Memory Limit registers indicate support for 64-bit addressing, then the Prefetchable Limit Upper 32 Bits register is implemented as a read/write register which must be initialized by configuration software. If a 64-bit prefetchable memory address range is supported, the Prefetchable Limit Upper 32 Bits register specifies the upper 32 bits, corresponding to Address[63:32], of the 64-bit limit addresses which specify the prefetchable memory address range. |
| 19:17 | RO | 0x0 | reserved |
| 16 | RO | 0x1 | PREF_MEM_LIMIT_DECODE Prefetchable Memory Limit Decode. This bit encodes whether or not the bridge supports 64-bit addresses. The value of PREF_MEM_LIMIT_DECODE indicates the following: 0b: Indicates that the bridge supports only 32 bit addresses 1b: Indicates that the bridge supports 64 bit addresses. Prefetchable Limit Upper 32 Bits registers holds the rest of the 64-bit prefetchable limit address. |
| 15:4 | RW | 0x000 | PREF_MEM_BASE Prefetchable Memory Base Address. If the Prefetchable Memory Base register indicates support for 32-bit addressing, then the Prefetchable Base Upper 32 Bits register is implemented as a read-only register that returns zero when read. If the Prefetchable Memory Base register indicates support for 64-bit addressing, then the Prefetchable Limit Upper 32 Bits register is implemented as a read/write register which must be initialized by configuration software. If a 64-bit prefetchable memory address range is supported, the Prefetchable Base Upper 32 Bits register specifies the upper 32 bits, corresponding to Address[63:32], of the 64-bit base addresses which specify the prefetchable memory address range. |
| 3:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | RO | 0x1 | <p>PREF_MEM_DECODE Prefetchable Memory Base Decode. This bit encodes whether or not the bridge supports 64-bit addresses. The value of PREF_MEM_DECODE indicates the following:</p> <ul style="list-style-type: none"> 0b: Indicates that the bridge supports only 32 bit addresses. 1b: Indicates that the bridge supports 64 bit addresses. <p>Prefetchable Base Upper 32 Bits registers holds the rest of the 64-bit prefetchable base address.</p> <p>Note: The access attributes of this field are as follows:</p> <ul style="list-style-type: none"> Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

TYPE1 PREF BASE UPPER REG

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | <p>PREF_MEM_BASE_UPPER Prefetchable Base Upper 32 Bit. If the Prefetchable Memory Base register indicates support for 32-bit addressing, then this register is implemented as read-only register that returns zero when read. If the Prefetchable Memory Base register indicate support for 64-bit addressing, then this register is implemented as read/write register which must be initialized by configuration software. This register specifies the upper 32 bits, corresponding to Address[63:32], of the 64-bit base addresses which specify the prefetchable memory address range.</p> <p>Note: The access attributes of this field are as follows:</p> <ul style="list-style-type: none"> Wire: No access. Dbi: PREF_MEM_LIMIT_PREF_MEM_BASE_REG.PREF_MEM_DECODE ? RW : RO |

TYPE1 PREF LIMIT UPPER REG

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x0 | <p>PREF_MEM_LIMIT_UPPER</p> <p>Prefetchable Limit Upper 32 Bit. If the Prefetchable Memory Limit register indicate support for 64-bit addressing, then this register is implemented as read/write register which must be initialized by configuration software. This register specifies the upper 32 bits, corresponding to Address[63:32], of the 64-bit base addresses which specify the prefetchable memory address range.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: No access.</p> <p>Dbi:</p> <p>PREF_MEM_LIMIT_PREF_MEM_BASE_REG.PREF_MEM_DECODE ?</p> <p>RW : RO</p> |

TYPE1 IO LIMIT UPPER IO BASE UPPER REG

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RO | 0x0000 | <p>IO_LIMIT_UPPER</p> <p>I/O Limit Upper 16 Bits. If the I/O Limit register indicates support for 16-bit I/O address decoding, then this register is implemented as a read-only register which return zero when read. If the I/O Limit register indicates support for 32-bit I/O addressing, then this register must be initialized by configuration software. If 32-bit I/O address decoding is supported, this register specifies the upper 16 bits, corresponding to Address[31:16], of the 32-bit limit address, that specify the I/O address range. See the PCI-to-PCI Bridge Architecture Specification for additional details).</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: No access.</p> <p>Dbi: SEC_STAT_IO_LIMIT_IO_BASE_REG.IO_DECODE ? RW : RO</p> |
| 15:0 | RO | 0x0000 | <p>IO_BASE_UPPER</p> <p>I/O Base Upper 16 Bits. If the I/O Base register indicates support for 16-bit I/O address decoding, then this register is implemented as a read-only register which return zero when read. If the I/O base register indicates support for 32-bit I/O addressing, then this register must be initialized by configuration software. If 32-bit I/O address decoding is supported, this register specifies the upper 16 bits, corresponding to Address[31:16], of the 32-bit base address, that specify the I/O address range. See the PCI-to-PCI Bridge Architecture Specification for additional details.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: No access.</p> <p>Dbi: SEC_STAT_IO_LIMIT_IO_BASE_REG.IO_DECODE ? RW : RO</p> |

TYPE1 TYPE1 CAP PTR REG

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RO | 0x40 | <p>CAP_POINTER Capabilities Pointer. This register is used to point to a linked list of capabilities implemented by this Function. Since all PCI Express Functions are required to implement the PCI Express Capability structure, this register must point to a valid capability structure and either this structure is the PCI Express Capability structure, or a subsequent list item points to the PCI Express Capability structure. The bottom two bits are Reserved and must be set to 00b. Software must mask these bits off before using this register as a pointer in Configuration Space to the first entry of a linked list of new capabilities.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |

TYPE1 TYPE1 EXP ROM BASE REG

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:11 | RO | 0x000000 | <p>EXP_ROM_BASE_ADDRESS Expansion ROM Base Address. Upper 21 bits of the Expansion ROM base address. The number of bits (out of these 21) that a Function actually implements depends on how much address space the Function requires. The mask for this ROM BAR exists (if implemented) as a shadow register at this address. The assertion of CS2 (that is, assert the dbi_cs2 input, or the CS2 address bit for the AXI bridge) is required to write to the second register at this address.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (EXP_ROM_BAR_MASK_REG.ROM_BAR_ENABLED == 1) then R/W else R</p> |
| 10:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | RO | 0x0 | <p>ROM_BAR_ENABLE Expansion ROM Enable. This bit controls whether or not the Function accepts accesses to its expansion ROM. When this bit is 0b, the Function's expansion ROM address space is disabled. When the bit is 1b, address decoding is enabled using the parameters in the other part of the Expansion ROM Base Address register. The Memory Space Enable bit in the Command register has precedence over the Expansion ROM Enable bit. A Function must claim accesses to its expansion ROM only if both the Memory Space Enable bit and the Expansion ROM Enable bit are set.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (EXP_ROM_BAR_MASK_REG.ROM_BAR_ENABLED == 1) then R/W else R</p> |

TYPE1 BRIDGE CTRL INT PIN INT LINE REG

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:23 | RO | 0x0 | reserved |
| 22 | RW | 0x0 | <p>SBR Secondary Bus Reset. Setting this bit triggers a hot reset on the corresponding PCI Express Port. Software must ensure a minimum reset duration (Trst) as defined in the PCI Local Bus Specification. Software and systems must honor first-access-following-reset timing requirements, unless the Readiness Notifications mechanism is used or if the Immediate Readiness bit in the relevant Function's Status Register register is set. Port configuration registers must not be changed, except as required to update Port status.</p> |
| 21 | RO | 0x0 | <p>MSTR_ABORT_MODE Master Abort Mode. This bit was originally described in the PCI-to-PCI Bridge Architecture Specification. Its functionality does not apply to PCI Express. The controller hardwires this bit to 0b.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: R/W</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 20 | RO | 0x0 | <p>VGA_16B_DEC</p> <p>VGA 16 bit decode. This bit only has meaning if VGA Enable bit is set. This bit enables system configuration software to select between 10-bit and 16-bit I/O address decoding for all VGA I/O register accesses that are forwarded from primary to secondary. The following actions are taken based on the value of the VGA_16B_DEC bit:</p> <ul style="list-style-type: none"> 0b: Execute 10-bit address decodes on VGA I/O accesses 1b: Execute 16-bit address decodes on VGA I/O accesses <p>For Functions that do not support VGA, the controller hardwires this bit to 0b.</p> <p>Note: The access attributes of this field are as follows:</p> <ul style="list-style-type: none"> Wire: No access. Dbi: R |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 19 | RO | 0x0 | <p>VGA_EN</p> <p>VGA Enable. Modifies the response by the bridge to VGA compatible addresses. If the VGA Enable bit is set, the bridge will positively decode and forward the following accesses on the primary interface to the secondary interface (and, conversely, block the forwarding of these addresses from the secondary to primary interface):</p> <p>Memory accesses in the range 000A 0000h to 000B FFFFh I/O addresses in the first 64 KB of the I/O address space (Address[31:16] are 0000h) where Address[9:0] are in the ranges 3B0h to 3BBh and 3C0h to 3DFh (inclusive of ISA address aliases determined by the setting of VGA 16-bit Decode)</p> <p>If the VGA Enable bit is set, forwarding of these accesses is independent of the I/O address range and memory address ranges defined by the I/O Base and Limit registers, the Memory Base and Limit registers, and the Prefetchable Memory Base and Limit registers of the bridge. (Forwarding of these accesses is also independent of the setting of the ISA Enable bit (in the Bridge Control register) when the VGA Enable bit is set.</p> <p>Forwarding of these accesses is qualified by the I/O Space Enable and Memory Space Enable bits in the Command register.) The following actions are taken based on the value of the VGA_EN bit:</p> <p>0b: Do not forward VGA compatible memory and I/O addresses from the primary to the secondary interface (addresses defined above) unless they are enabled for forwarding by the defined I/O and memory address ranges</p> <p>1b: Forward VGA compatible memory and I/O addresses (addresses defined above) from the primary interface to the secondary interface (if the I/O Space Enable and Memory Space Enable bits are set) independent of the I/O and memory address ranges and independent of the ISA Enable bit</p> <p>For Functions that do not support VGA, the controller hardwires this bit to 0b.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: No access.</p> <p>Dbi: R</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 18 | RW | 0x0 | <p>ISA_EN</p> <p>ISA Enable. Modifies the response by the bridge to ISA I/O addresses. This applies only to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64 KB of I/O address space (0000 0000h to 0000 FFFFh). If this bit is set, the bridge will block any forwarding from primary to secondary of I/O transactions addressing the last 768 bytes in each 1-KB block. In the opposite direction (secondary to primary), I/O transactions will be forwarded if they address the last 768 bytes in each 1-KB block. The following actions are taken based on the value of the ISA_EN bit:</p> <p>0b: Forward downstream all I/O addresses in the address range defined by the I/O Base and I/O Limit registers</p> <p>1b: Forward upstream ISA I/O addresses in the address range defined by the I/O Base and I/O Limit registers that are in the first 64 KB of PCI I/O address space (top 768 bytes of each 1-KB block).</p> |
| 17 | RW | 0x0 | <p>SERR_EN</p> <p>SERR# Enable. This bit controls forwarding of ERR_COR, ERR_NONFATAL and ERR_FATAL from secondary to primary.</p> |
| 16 | RW | 0x0 | <p>PERE</p> <p>Parity Error Response Enable. This bit controls the logging of poisoned TLPs in the Master Data Parity Error bit in the Secondary Status register.</p> |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 15:8 | RO | 0x01 | <p>INT_PIN Interrupt PIN. The Interrupt Pin register identifies the legacy interrupt Message(s) the Function uses. Valid values are: 01h, 02h, 03h, and 04h: map to legacy interrupt Messages for INTA, INTB, INTC, and INTD respectively. 00h: indicates that the Function uses no legacy interrupt Message(s). 05h through FFh: Reserved. PCI Express defines one legacy interrupt Message for a single Function device and up to four legacy interrupt Messages for a multi-Function device. For a single Function device, only INTA may be used. Any Function on a multi-Function device can use any of the INTx Messages. If a device implements a single legacy interrupt Message, it must be INTA; if it implements two legacy interrupt Messages, they must be INTA and INTB; and so forth. For a multi-Function device, all Functions may use the same INTx Message or each may have its own (up to a maximum of four Functions) or any combination thereof. A single Function can never generate an interrupt request on more than one INTx Message.</p> <p>Note: The access attributes of this field are as follows: Wire: No access. Dbi: if (DBI_RO_WR_EN == 1) then R/W else R Note: This register field is sticky.</p> |
| 7:0 | RW | 0xff | <p>INT_LINE Interrupt Line. The Interrupt Line register communicates interrupt line routing information. The register must be implemented by any Function that uses an interrupt pin. Values in this register are programmed by system software and are system architecture specific. The Function itself does not use this value; rather the value in this register is used by device drivers and operating systems.</p> |

10.5.8 Power Management Capability Registers Summary

PF PCI Power Management Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|-----------------------|--------|------|-------------|---|
| PM_CAP_ID_NXT_PTR_REG | 0x0040 | W | 0x07c35001 | Power Management Capabilities Register. |
| PM_CON_STATUS_REG | 0x0044 | W | 0x00000008 | Power Management Control and Status Register. |

Notes: **S**-Size: **B**- Byte (8 bits) access, **H**W- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.9 Power Management Capability Detail Register Description

PM_CAP_ID_NXT_PTR_REG

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x00 | PME_SUPPORT Power Management Event Support. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 26 | RO | 0x1 | D2_SUPPORT D2 State Support. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 25 | RO | 0x1 | D1_SUPPORT D1 State Support. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 24:22 | RO | 0x7 | AUX_CURR Auxiliary Current Requirements. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 21 | RO | 0x0 | DSI Device Specific Initialization Bit. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 20 | RO | 0x0 | reserved |
| 19 | RO | 0x0 | PME_CLK PCI Clock Requirement. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18:16 | RO | 0x3 | PM_SPEC_VER Power Management Spec Version. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 15:8 | RO | 0x50 | PM_NEXT_POINTER Next Capability Pointer. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 7:0 | RW | 0x01 | PM_CAP_ID Power Management Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. |

PM CON STATUS REG

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | DATA_REG_ADD_INFO Power Data Information Register. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23 | RO | 0x0 | BUS_PWR_CLK_CON_EN Bus Power/Clock Control Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 22 | RO | 0x0 | B2_B3_SUPPORT B2B3 Support for D3hot. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 21:16 | RO | 0x0 | reserved |
| 15 | RW | 0x0 | PME_STATUS PME Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 14:13 | RO | 0x0 | DATA_SCALE Data Scaling Factor. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12:9 | RO | 0x0 | DATA_SELECT Data Select. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 8 | RW | 0x0 | PME_ENABLE PME Enable. For a description of this standard PCIe register field, see the PCI Express Specification. The PMC registers this value under aux power. Sometimes it might remember the old value, even if you try to clear it by writing '0'. |
| 7:4 | RO | 0x0 | reserved |
| 3 | RO | 0x1 | NO_SOFT_RST No soft Reset. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x0 | POWER_STATE Power State. For a description of this standard PCIe register field, see the PCI Express Specification. You can write to this register. However, the read-back value is the actual power state, not the write value. |

10.5.10 MSI Capability Registers Summary

PF MSI Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|---------------------------------|--------|------|-------------|--|
| <u>MSI_CAP_ID_NEXT_CTRL_REG</u> | 0x0050 | W | 0x018a7005 | MSI Capability ID, Next Pointer, Capability/Control Registers. |
| <u>MSI_CAP_OFF_04H_REG</u> | 0x0054 | W | 0x00000000 | MSI Message Lower Address Register. |
| <u>MSI_CAP_OFF_08H_REG</u> | 0x0058 | W | 0x00000000 | For a 32 bit MSI Message, this register contains Data. For 64 bit it contains the Upper Address. |
| <u>MSI_CAP_OFF_0CH_REG</u> | 0x005c | W | 0x00000000 | For a 64 bit MSI Message, this register contains Data. For 32 bit, it contains Mask Bits if PVM enabled. |
| <u>MSI_CAP_OFF_10H_REG</u> | 0x0060 | W | 0x00000000 | For 32 bit contains Pending Bits. For 64 bit, contains Mask Bits. |
| <u>MSI_CAP_OFF_14H_REG</u> | 0x0064 | W | 0x00000000 | Contains Pending Bits. |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.11 MSI Capability Detail Register Description

MSI_CAP_ID_NEXT_CTRL_REG

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:27 | RO | 0x0 | reserved |
| 26 | RO | 0x0 | PCI_MSI_EXT_DATA_EN Extended Message Data Enable. For a description of this standard PCIe register, see the PCI-SIG ECN for Extended MSI Data, Feb 24, 2016, affecting PCI Express Specification. |
| 25 | RO | 0x0 | PCI_MSI_EXT_DATA_CAP Extended Message Data Capable. For a description of this standard PCIe register, see the PCI-SIG ECN for Extended MSI Data, Feb 24, 2016, affecting PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 24 | RO | 0x1 | PCI_PVM_SUPPORT MSI Per Vector Masking Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23 | RO | 0x1 | PCI_MSI_64_BIT_ADDR_CAP MSI 64-bit Address Capable. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 22:20 | RW | 0x0 | PCI_MSI_MULTIPLE_MSG_EN MSI Multiple Message Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 19:17 | RO | 0x5 | PCI_MSI_MULTIPLE_MSG_CAP MSI Multiple Message Capable. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 16 | RW | 0x0 | PCI_MSI_ENABLE MSI Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x70 | PCI_MSI_CAP_NEXT_OFFSET MSI Capability Next Pointer. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 7:0 | RO | 0x05 | PCI_MSI_CAP_ID MSI Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. |

MSI CAP OFF 04H REG

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:2 | RW | 0x00000000 | PCI_MSI_CAP_OFF_04H MSI Message Lower Address Field. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1:0 | RO | 0x0 | reserved |

MSI CAP OFF 08H REG

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | PCI_MSI_CAP_OFF_0AH For a 32 bit MSI Message, this field contains Ext MSI Data. For 64-bit it contains upper 16 bits of the Upper Address. For a description of this standard PCIe register field, see the PCI Express Specification. Note: writeable when PCI_MSI_64_BIT_ADDR_CAP = 1. |
| 15:0 | RW | 0x0000 | PCI_MSI_CAP_OFF_08H For a 32-bit MSI Message, this field contains Data. For 64-bit it contains lower 16 bits of the Upper Address. For a description of this standard PCIe register field, see the PCI Express Specification. Note: writeable when PCI_MSI_64_BIT_ADDR_CAP = 1. |

MSI CAP OFF 0CH REG

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RW | 0x0000 | PCI_MSI_CAP_OFF_0EH For a 64-bit MSI Message, this field contains Data. For 32-bit, it contains the upper Mask Bits if PVM is enabled. For a description of this standard PCIe register field, see the PCI Express Specification |
| 15:0 | RW | 0x0000 | PCI_MSI_CAP_OFF_0CH For a 64-bit MSI Message, this field contains Data. For 32-bit, it contains the lower Mask Bits if PVM is enabled. For a description of this standard PCIe register field, see the PCI Express Specification. |

MSI CAP OFF 10H REG

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | PCI_MSI_CAP_OFF_10H Used for MSI when the Per Vector Masking Capable bit (PCI_MSI_CAP_ID_NEXT_CTRL_REG.PCI_PVM_SUPPORT) is set. For 32-bit contains Pending Bits. For 64-bit, contains Mask Bits. For a description of this standard PCIe register field, see the PCI Express Specification. |

MSI CAP OFF 14H REG

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | W1C | 0x0 | PCI_MSI_CAP_OFF_14H Used for MSI 64-bit messaging when the Per Vector Masking Capable bit (PCI_MSI_CAP_ID_NEXT_CTRL_REG.PCI_PVM_SUPPORT) is set. Contains Pending Bits. For a description of this standard PCIe register field, see the PCI Express Specification. |

10.5.12 PCIE Capability Registers Summary

PF PCI Express Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|---|--------|------|-------------|---|
| PCIE CAP ID PCIE NEXT CAP PTR PCIE CAP REG | 0x0070 | W | 0x1002b010 | PCI Express Capabilities, ID, Next Pointer Register. |
| PCIE CAP DEVICE CAPABILITIES REG | 0x0074 | W | 0x00008fc0 | Device Capabilities Register. |
| PCIE CAP DEVICE CONTROL DEVICE STATUS | 0x0078 | W | 0x00002010 | Device Control and Status Register. |
| PCIE CAP LINK CAPABILITIES REG | 0x007c | W | 0x00437c23 | Link Capabilities Register. |
| PCIE CAP LINK CONTROL LINK STATUS REG | 0x0080 | W | 0x10000000 | Link Control and Status Register. |
| PCIE CAP SLOT CAPABILITIES REG | 0x0084 | W | 0x00000000 | Slot Capabilities Register. Exists only in RC mode. |
| PCIE CAP SLOT CONTROL SLOT STATUS | 0x0088 | W | 0x000003c0 | Slot Control and Status Register. Exists only in RC mode. |
| PCIE CAP ROOT CONTROL ROOT CAPABILITIES REG | 0x008c | W | 0x00010000 | Root Control and Capabilities Register. Exists only in RC mode. |
| PCIE CAP ROOT STATUS REG | 0x0090 | W | 0x00000000 | Root Status Register. Exists only in RC mode. |
| PCIE CAP DEVICE CAPABILITIES2 REG | 0x0094 | W | 0x000c0810 | Device Capabilities 2 Register. |
| PCIE CAP DEVICE CONTROL2 DEVICE STATUS2 REG | 0x0098 | W | 0x00000000 | Device Control 2 and Status 2 Register. |
| PCIE CAP LINK CAPABILITIES2 REG | 0x009c | W | 0x0000000e | Link Capabilities 2 Register. |
| PCIE CAP LINK CONTROL2 LINK STATUS2 REG | 0x00a0 | W | 0x00010003 | Link Control 2 and Status 2 Register. |

Notes: **S**-ize: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.13 PCIE Capability Detail Register Description

PCIE CAP ID PCIE NEXT CAP PTR PCIE CAP REG

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29:25 | RO | 0x08 | PCIE_INT_MSG_NUM PCIE Interrupt Message Number. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 24 | RO | 0x0 | PCIE_SLOT_IMP PCIe Slot Implemented Valid. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 23:20 | RO | 0x0 | PCIE_DEV_PORT_TYPE PCIe Device/PortType. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19:16 | RO | 0x2 | PCIE_CAP_REG PCIe Capability Version Number. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0xb0 | PCIE_CAP_NEXT_PTR PCIe Next Capability Pointer. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 7:0 | RO | 0x10 | PCIE_CAP_ID PCIe Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP DEVICE CAPABILITIES REG

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28 | RO | 0x0 | PCIE_CAP_FLR_CAP Function Level Reset Capability (endpoints only). For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 27:26 | RO | 0x0 | PCIE_CAP_CAP_SLOT_PWR_LMT_SCALE Captured Slot Power Limit Scale. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 25:18 | RO | 0x00 | PCIE_CAP_CAP_SLOT_PWR_LMT_VALUE Captured Slot Power Limit Value. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 17:16 | RO | 0x0 | reserved |
| 15 | RO | 0x1 | PCIE_CAP_ROLE_BASED_ERR_REPORT Role-based Error Reporting Implemented. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 14:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:9 | RO | 0x7 | PCIE_CAP_EP_L1_ACCPT_LATENCY Applies to endpoints only L1 acceptable latency. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 8:6 | RO | 0x7 | PCIE_CAP_EP_L0S_ACCPT_LATENCY Applies to endpoints only L0s acceptable latency. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 5 | RO | 0x0 | PCIE_CAP_EXT_TAG_SUPP Extended Tag Field Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 4:3 | RO | 0x0 | PCIE_CAP_PHANTOM_FUNC_SUPPORT Phantom Functions Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 2:0 | RO | 0x0 | PCIE_CAP_MAX_PAYLOAD_SIZE Max Payload Size Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

PCIE CAP DEVICE CONTROL DEVICE STATUS

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:22 | RO | 0x0 | reserved |
| 21 | RO | 0x0 | PCIE_CAP_TRANS_PENDING Transactions Pending Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 20 | RO | 0x0 | PCIE_CAP_AUX_POWER_DETECTED Aux Power Detected Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19 | RW | 0x0 | PCIE_CAP_UNSUPPORTED_REQ_DETECTED Unsupported Request Detected Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RO | 0x0 | reserved |
| 17 | RW | 0x0 | PCIE_CAP_NON_FATAL_ERR_DETECTED Non-Fatal Error Detected Status. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16:15 | RO | 0x0 | reserved |
| 14:12 | RW | 0x2 | PCIE_CAP_MAX_READ_REQ_SIZE Max Read Request Size. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11 | RO | 0x0 | PCIE_CAP_EN_NO_SNOOP Enable No Snoop. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 10 | RW | 0x0 | PCIE_CAP_AUX_POWER_PM_EN Aux Power PM Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 9 | RO | 0x0 | PCIE_CAP_PHANTOM_FUNC_EN Phantom Functions Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 8 | RO | 0x0 | PCIE_CAP_EXT_TAG_EN Extended Tag Field Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:5 | RW | 0x0 | PCIE_CAP_MAX_PAYLOAD_SIZE_CS Max Payload Size. Max_Payload_Size . This field sets maximum TLP payload size for the Function. Permissible values that can be programmed are indicated by the Max_Payload_Size Supported field (PCIE_CAP_MAX_PAYLOAD_SIZE) in the Device Capabilities register (DEVICE_CAPABILITIES_REG). |
| 4 | RW | 0x1 | PCIE_CAP_EN_REL_ORDER Enable Relaxed Ordering. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3 | RW | 0x0 | PCIE_CAP_UNSUPPORT_REQ REP_EN Unsupported Request Reporting Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RW | 0x0 | PCIE_CAP_FATAL_ERR_REPORT_EN Fatal Error Reporting Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1 | RW | 0x0 | PCIE_CAP_NON_FATAL_ERR_REPORT_EN Non-fatal Error Reporting Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RW | 0x0 | PCIE_CAP_CORR_ERR_REPORT_EN Correctable Error Reporting Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP LINK CAPABILITIES REG

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | PCIE_CAP_PORT_NUM Port Number. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 23 | RO | 0x0 | reserved |
| 22 | RO | 0x1 | PCIE_CAP_ASPM_OPT_COMPLIANCE ASPM Optionality Compliance. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 21 | RO | 0x0 | PCIE_CAP_LINK_BW_NOT_CAP Link Bandwidth Notification Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 20 | RO | 0x0 | PCIE_CAP_DLL_ACTIVE REP_CAP Data Link Layer Link Active Reporting Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19 | RO | 0x0 | PCIE_CAP_SURPRISE_DOWN_ERR REP_CAP Surprise Down Error Reporting Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RO | 0x0 | PCIE_CAP_CLOCK_POWER_MAN Clock Power Management. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 17:15 | RO | 0x6 | <p>PCIE_CAP_L1_EXIT_LATENCY L1 Exit Latency. For a description of this standard PCIe register field, see the PCI Express Specification. There are two each of these register fields, this one and a shadow one at the same address. The Common Clock bit (PCIE_CAP_COMMON_CLK_CONFIG) of the Link Control Register (LINK_CONTROL_LINK_STATUS_REG) determines which one is used by the controller and which one is accessed by a read request. Common Clock operation is supported (possible) in the controller when one or more of the following expressions is true:</p> <p>CX_NFTS != CX_COMM_NFTS DEFAULT_L0S_EXIT_LATENCY != DEFAULT_COMM_L0S_EXIT_LATENCY DEFAULT_L1_EXIT_LATENCY != DEFAULT_COMM_L1_EXIT_LATENCY</p> <p>Common Clock operation is enabled in the controller when you set the Common Clock bit (PCIE_CAP_COMMON_CLK_CONFIG) of the Link Control Register (LINK_CONTROL_LINK_STATUS_REG). The assertion of CS2 (that is, assert the dbi_cs2 input, or the CS2 address bit for the AXI bridge) is required to write to the shadow field at this location.</p> |
| 14:12 | RO | 0x7 | <p>PCIE_CAP_L0S_EXIT_LATENCY L0S Exit Latency. For a description of this standard PCIe register field, see the PCI Express Specification. There are two each of these register fields, this one and a shadow one at the same address. The Common Clock bit (PCIE_CAP_COMMON_CLK_CONFIG) of the Link Control Register (LINK_CONTROL_LINK_STATUS_REG) determines which one is used by the controller and which one is accessed by a read request. Common Clock operation is supported (possible) in the controller when one or more of the following expressions is true:</p> <p>CX_NFTS != CX_COMM_NFTS DEFAULT_L0S_EXIT_LATENCY != DEFAULT_COMM_L0S_EXIT_LATENCY DEFAULT_L1_EXIT_LATENCY != DEFAULT_COMM_L1_EXIT_LATENCY</p> <p>Common Clock operation is enabled in the controller when you set the Common Clock bit (PCIE_CAP_COMMON_CLK_CONFIG) of the Link Control Register (LINK_CONTROL_LINK_STATUS_REG). The assertion of CS2 (that is, assert the dbi_cs2 input, or the CS2 address bit for the AXI bridge) is required to write to the shadow field at this location.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:10 | RO | 0x3 | PCIE_CAP_ACTIVE_STATE_LINK_PM_SUPPORT xLevel of ASPM (Active State Power Management) Support. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 9:4 | RO | 0x02 | PCIE_CAP_MAX_LINK_WIDTH Maximum Link Width. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 3:0 | RO | 0x3 | PCIE_CAP_MAX_LINK_SPEED Maximum Link Speed. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

PCIE CAP LINK CONTROL LINK STATUS REG

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | PCIE_CAP_LINK_AUTO_BW_STATUS Link Autonomous Bandwidth Status. For a description of this standard PCIe register field, see the PCI Express Specification. The write value is gated with the PCIE_CAP_LINK_BW_NOT_CAP field in LINK_CAPABILITIES_REG. |
| 30 | RO | 0x0 | PCIE_CAP_LINK_BW_MAN_STATUS Link Bandwidth Management Status. For a description of this standard PCIe register field, see the PCI Express Specification. The write value is gated with the PCIE_CAP_LINK_BW_NOT_CAP field in LINK_CAPABILITIES_REG. |
| 29 | RO | 0x0 | PCIE_CAP_DLL_ACTIVE Data Link Layer Active. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 28 | RO | 0x1 | PCIE_CAP_SLOT_CLK_CONFIG Slot Clock Configuration. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 27 | RO | 0x0 | PCIE_CAP_LINK_TRAINING LTSSM is in Configuration or Recovery State. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 26 | RO | 0x0 | reserved |
| 25:20 | RO | 0x00 | PCIE_CAP_NEGO_LINK_WIDTH Negotiated Link Width. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 19:16 | RO | 0x0 | PCIE_CAP_LINK_SPEED Current Link Speed. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:14 | RO | 0x0 | PCIE_CAP_DRS_SIGNALING_CONTROL DRS Signaling Control. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 13:12 | RO | 0x0 | reserved |
| 11 | RW | 0x0 | PCIE_CAP_LINK_AUTO_BW_INT_EN Link Autonomous Bandwidth Management Interrupt Enable. For a description of this standard PCIe register field, see the PCI Express Specification. The write value is gated with the PCIE_CAP_LINK_BW_NOT_CAP field in LINK_CAPABILITIES_REG. |
| 10 | RW | 0x0 | PCIE_CAP_LINK_BW_MAN_INT_EN Link Bandwidth Management Interrupt Enable. For a description of this standard PCIe register field, see the PCI Express Specification. The write value is gated with the PCIE_CAP_LINK_BW_NOT_CAP field in LINK_CAPABILITIES_REG. |
| 9 | RW | 0x0 | PCIE_CAP_HW_AUTO_WIDTH_DISABLE Hardware Autonomous Width Disable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 8 | RW | 0x0 | PCIE_CAP_EN_CLK_POWER_MAN Enable Clock Power Management. For a description of this standard PCIe register field, see the PCI Express Specification. The write value is gated with the PCIE_CAP_CLOCK_POWER_MAN field in LINK_CAPABILITIES_REG. |
| 7 | RW | 0x0 | PCIE_CAP_EXTENDED_SYNCH Extended Synch. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 6 | RW | 0x0 | PCIE_CAP_COMMON_CLK_CONFIG Common Clock Configuration. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5 | RW | 0x0 | PCIE_CAP_RETRAIN_LINK Initiate Link Retrain. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RW | 0x0 | PCIE_CAP_LINK_DISABLE Initiate Link Disable. For a description of this standard PCIe register field, see the PCI Express Specification. In a DSP that supports crosslink, the controller gates the write value with the CROSS_LINK_EN field in PORT_LINK_CTRL_OFF. |
| 3 | RW | 0x0 | PCIE_CAP_RCB Read Completion Boundary (RCB). |
| 2 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 1:0 | RW | 0x0 | PCIE_CAP_ACTIVE_STATE_LINK_PM_CONTROL Active State Power Management (ASPM) Control. Software must not enable L0s in either direction on a given Link unless components on both sides of the Link each support L0s; otherwise, the result is undefined. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP SLOT CAPABILITIES REG

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:19 | RO | 0x0000 | PCIE_CAP_PHY_SLOT_NUM Physical Slot Number. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RO | 0x0 | PCIE_CAP_NO_CMD_CPL_SUPPORT No Command Completed Support. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 17 | RO | 0x0 | PCIE_CAP_ELECTROMECH_INTERLOCK Electromechanical Interlock Present. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 16:15 | RO | 0x0 | PCIE_CAP_SLOT_POWER_LIMIT_SCALE Slot Power Limit Scale. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 14:7 | RO | 0x00 | PCIE_CAP_SLOT_POWER_LIMIT_VALUE Slot Power Limit Value. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 6 | RO | 0x0 | PCIE_CAP_HOT_PLUG_CAPABLE Hot Plug Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5 | RO | 0x0 | PCIE_CAP_HOT_PLUG_SURPRISE Hot Plug Surprise possible. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RO | 0x0 | PCIE_CAP_POWER_INDICATOR Power Indicator Present. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3 | RO | 0x0 | PCIE_CAP_ATTENTION_INDICATOR Attention Indicator Present. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RO | 0x0 | PCIE_CAP_MRL_SENSOR MRL Present. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1 | RO | 0x0 | PCIE_CAP_POWER_CONTROLLER Power Controller Present. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 0 | RO | 0x0 | PCIE_CAP_ATTENTION_INDICATOR_BUTTON Attention Button Present. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP SLOT CONTROL SLOT STATUS

Address: Operational Base + offset (0x0088)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | PCIE_CAP_DLL_STATE_CHANGED Data Link Layer State Changed. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23 | RO | 0x0 | PCIE_CAP_ELECTROMECH_INTERLOCK_STATUS Electromechanical Interlock Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 22 | RO | 0x0 | PCIE_CAP_PRESENCE_DETECT_STATE Presence Detect State. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 21 | RO | 0x0 | PCIE_CAP_MRL_SENSOR_STATE MRL Sensor State. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 20 | RW | 0x0 | PCIE_CAP_CMD_CPLD Command Completed. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19 | RW | 0x0 | PCIE_CAP_PRESENCE_DETECTED_CHANGED Presence Detect Changed. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RW | 0x0 | PCIE_CAP_MRL_SENSOR_CHANGED MRL Sensor Changed. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 17 | RW | 0x0 | PCIE_CAP_POWER_FAULT_DETECTED Power Fault Detected. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 16 | RW | 0x0 | PCIE_CAP_ATTENTION_BUTTON_PRESSED Attention Button Pressed. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:13 | RO | 0x0 | reserved |
| 12 | RW | 0x0 | PCIE_CAP_DLL_STATE_CHANGED_EN Data Link Layer State Changed Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11 | RW | 0x0 | PCIE_CAP_ELECTROMECH_INTERLOCK_CTRL Electromechanical Interlock Control. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 10 | RW | 0x0 | PCIE_CAP_POWER_CONTROLLER_CTRL Power Controller Control. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 9:8 | RW | 0x3 | PCIE_CAP_POWER_INDICATOR_CTRL Power Indicator Control. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:6 | RW | 0x3 | PCIE_CAP_ATTENTION_INDICATOR_CTRL Attention Indicator Control. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5 | RW | 0x0 | PCIE_CAP_HOT_PLUG_INT_EN Hot Plug Interrupt Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RW | 0x0 | PCIE_CAP_CMD_CPL_INT_EN Command Completed Interrupt Enable. For a description of this standard PCIe register field, see the PCI Express Specification. Write value is gated with PCIE_CAP_NO_CMD_CPL_SUPPORT field in SLOT_CAPABILITIES_REG. |
| 3 | RW | 0x0 | PCIE_CAP_PRESENCE_DETECT_CHANGE_EN Presence Detect Changed Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RW | 0x0 | PCIE_CAP_MRL_SENSOR_CHANGED_EN MRL Sensor Changed Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1 | RW | 0x0 | PCIE_CAP_POWER_FAULT_DETECTED_EN Power Fault Detected Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RW | 0x0 | PCIE_CAP_ATTENTION_BUTTON_PRESSED_EN Attention Button Pressed Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP ROOT CONTROL ROOT CAPABILITIES REG

Address: Operational Base + offset (0x008c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16 | RO | 0x1 | PCIE_CAP_CRS_SW_VISIBILITY CRS Software Visibility Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:5 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 4 | RW | 0x0 | PCIE_CAP CRS SW VISIBILITY_EN Configuration Request Retry Status (CRS) Software Visibility Enable. For a description of this standard PCIe register field, see the PCI Express Specification. Writeable when ROOT_CONTROL_ROOT_CAPABILITIES_REG.PCIE_CAP CRS SW _VISIBILITY=1. |
| 3 | RW | 0x0 | PCIE_CAP PME INT EN PME Interrupt Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RW | 0x0 | PCIE_CAP SYS ERR ON FATAL_ERR_EN System Error on Fatal Error Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1 | RW | 0x0 | PCIE_CAP SYS ERR ON NON_FATAL_ERR_EN System Error on Non-fatal Error Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RW | 0x0 | PCIE_CAP SYS ERR ON CORR_ERR_EN System Error on Correctable Error Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP ROOT STATUS REG

Address: Operational Base + offset (0x0090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:18 | RO | 0x0 | reserved |
| 17 | RO | 0x0 | PCIE_CAP PME PENDING PME Pending. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 16 | RW | 0x0 | PCIE_CAP PME STATUS PME Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:0 | RO | 0x0000 | PCIE_CAP PME REQ_ID PME Requester ID. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP DEVICE CAPABILITIES2 REG

Address: Operational Base + offset (0x0094)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:18 | RO | 0x3 | PCIE_CAP OBFF SUPPORT (OBFF) Optimized Buffer Flush/fill Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17 | RO | 0x0 | PCIE_CAP2_10_BIT_TAG_REQ_SUPPORT 10-Bit Tag Requester Supported. For a description of this standard PCIe register field, see the PCI Express Base Specification 4.0. |
| 16 | RO | 0x0 | PCIE_CAP2_10_BIT_TAG_COMP_SUPPORT 10-Bit Tag Completer Supported. For a description of this standard PCIe register field, see the PCI Express Base Specification 4.0. |
| 15:14 | RO | 0x0 | PCIE_CAP2_LN_SYS_CLS LN System CLS. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 13 | RO | 0x0 | PCIE_CAP_TPH_CMPLT_SUPPORT_1 TPH Completer Supported Bit 1. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12 | RO | 0x0 | PCIE_CAP_TPH_CMPLT_SUPPORT_0 TPH Completer Supported Bit 0. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11 | RO | 0x1 | PCIE_CAP_LTR_SUPP LTR Mechanism Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 10 | RO | 0x0 | PCIE_CAP_NO_RO_EN_PR2PR_PAR No Relaxed Ordering Enabled PR-PR Passing. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 9 | RO | 0x0 | PCIE_CAP_128_CAS_CPL_SUPP 128 Bit CAS Completer Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 8 | RO | 0x0 | PCIE_CAP_64_ATOMIC_CPL_SUPP 64 Bit AtomicOp Completer Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7 | RO | 0x0 | PCIE_CAP_32_ATOMIC_CPL_SUPP 32 Bit AtomicOp Completer Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 6 | RO | 0x0 | PCIE_CAP_ATOMIC_ROUTING_SUPP Atomic Operation Routing Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5 | RO | 0x0 | PCIE_CAP_ARI_FORWARD_SUPPORT ARI Forwarding Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RO | 0x1 | PCIE_CAP_CPL_TIMEOUT_DISABLE_SUPPORT Completion Timeout Disable Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3:0 | RO | 0x0 | PCIE_CAP_CPL_TIMEOUT_RANGE Completion Timeout Ranges Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP DEVICE CONTROL2 DEVICE STATUS2 REG

Address: Operational Base + offset (0x0098)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:15 | RO | 0x0 | reserved |
| 14:13 | RW | 0x0 | PCIE_CAP_OBFF_EN OBFF Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12:11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | PCIE_CAP_LTR_EN LTR Mechanism Enable. For a description of this standard PCIe register field, see the PCI Express Specification. The write value is gated with the PCIE_CAP_LTR_SUPP field of DEVICE_CAPABILITIES2_REG. |
| 9:6 | RO | 0x0 | reserved |
| 5 | RO | 0x0 | PCIE_CAP_ARI_FORWARD_SUPPORT_CS ARI Forwarding Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RW | 0x0 | PCIE_CAP_CPL_TIMEOUT_DISABLE Completion Timeout Disable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3:0 | RO | 0x0 | PCIE_CAP_CPL_TIMEOUT_VALUE Completion Timeout Value. For a description of this standard PCIe register field, see the PCI Express Specification. |

PCIE CAP LINK CAPABILITIES2 REG

Address: Operational Base + offset (0x009c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:9 | RO | 0x0 | reserved |
| 8 | RO | 0x0 | PCIE_CAP_CROSS_LINK_SUPPORT Cross Link Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:1 | RO | 0x07 | PCIE_CAP_SUPPORT_LINK_SPEED_VECTOR Supported Link Speeds Vector. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RO | 0x0 | reserved |

PCIE CAP LINK CONTROL2 LINK STATUS2 REG

Address: Operational Base + offset (0x00a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30:28 | RO | 0x0 | DOWNSTREAM_COMPO_PRESENCE Downstream Component Presence. For a description of this standard PCIe register field, see the PCI Express Base Specification 4.0. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 27:22 | RO | 0x0 | reserved |
| 21 | RW | 0x0 | PCIE_CAP_LINK_EQ_REQ Link Equalization Request 8.0GT/s. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 20 | RO | 0x0 | PCIE_CAP_EQ_CPL_P3 Equalization 8.0GT/s Phase 3 Successful. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19 | RO | 0x0 | PCIE_CAP_EQ_CPL_P2 Equalization 8.0GT/s Phase 2 Successful. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RO | 0x0 | PCIE_CAP_EQ_CPL_P1 Equalization 8.0GT/s Phase 1 Successful. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 17 | RO | 0x0 | PCIE_CAP_EQ_CPL Equalization 8.0GT/s Complete. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 16 | RO | 0x1 | PCIE_CAP_CURR_DEEMPHASIS Current De-emphasis Level. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:12 | RW | 0x0 | PCIE_CAP_COMPLIANCE_PRESET Sets Compliance Preset/De-emphasis for 5 GT/s and 8 GT/s. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11 | RW | 0x0 | PCIE_CAP_COMPLIANCE_SOS Sets Compliance Skip Ordered Sets transmission. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 10 | RW | 0x0 | PCIE_CAP_ENTER_MODIFIED_COMPLIANCE Enter Modified Compliance. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 9:7 | RW | 0x0 | PCIE_CAP_TX_MARGIN Controls Transmit Margin for Debug or Compliance. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 6 | RO | 0x0 | PCIE_CAP_SEL_DEEMPHASIS Controls Selectable De-emphasis for 5 GT/s. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5 | RW | 0x0 | PCIE_CAP_HW_AUTO_SPEED_DISABLE Hardware Autonomous Speed Disable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RW | 0x0 | PCIE_CAP_ENTER_COMPLIANCE Enter Compliance Mode. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 3:0 | RW | 0x3 | PCIE_CAP_TARGET_LINK_SPEED Target Link Speed. For a description of this standard PCIe register field, see the PCI Express Specification. |

10.5.14 MSIX Capability Registers Summary

PF MSI-X Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|---------------------------------------|--------|------|-------------|---|
| MSIX PCI MSIX CAP ID NEXT CTRL REG | 0x00b0 | W | 0x003f0011 | MSI-X Capability ID, Next Pointer, Control Registers. |
| MSIX MSIX TABLE OFFS ET REG | 0x00b4 | W | 0x00000004 | MSI-X Table Offset and BIR Register. |
| MSIX MSIX PBA OFFSET REG | 0x00b8 | W | 0x00002804 | MSI-X PBA Offset and BIR Register. |

Notes: **S**-ize: **B**- Byte (8 bits) access, **H**W- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.15 MSIX Capability Detail Register Description

MSIX PCI MSIX CAP ID NEXT CTRL REG

Address: Operational Base + offset (0x00b0)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31 | RW | 0x0 | PCI_MSIX_ENABLE MSI-X Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 30 | RW | 0x0 | PCI_MSIX_FUNCTION_MASK Function Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 29:27 | RO | 0x0 | reserved |
| 26:16 | RO | 0x03f | PCI_MSIX_TABLE_SIZE MSI-X Table Size. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 15:8 | RO | 0x00 | PCI_MSIX_CAP_NEXT_OFFSET MSI-X Next Capability Pointer. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 7:0 | RO | 0x11 | PCI_MSIX_CAP_ID MSI-X Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. |

MSIX MSIX TABLE OFFSET REG

Address: Operational Base + offset (0x00b4)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:3 | RO | 0x00000000 | PCI_MSIX_TABLE_OFFSET MSI-X Table Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 2:0 | RO | 0x4 | PCI_MSIX_BIR MSI-X Table Bar Indicator Register Field. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

MSIX MSIX PBA OFFSET REG

Address: Operational Base + offset (0x00b8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:3 | RO | 0x00000500 | PCI_MSIX_PBA_OFFSET MSI-X PBA Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 2:0 | RO | 0x4 | PCI_MSIX_PBA MSI-X PBA BIR. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

10.5.16 AER Capability Registers Summary

PF Advanced Error Reporting Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|----------------------------|--------|------|-------------|---|
| AER_AER_EXT_CAP_HDR_OFF | 0x0100 | W | 0x14820001 | Advanced Error Reporting Extended Capability Header. |
| AER_UNCORR_ERR_STAT_US_OFF | 0x0104 | W | 0x00000000 | Uncorrectable Error Status Register. |
| AER_UNCORR_ERR_MASK_OFF | 0x0108 | W | 0x00400000 | Uncorrectable Error Mask Register. |
| AER_UNCORR_ERR_SEV_OFF | 0x010c | W | 0x00462030 | Uncorrectable Error Severity Register. |
| AER_CORR_ERR_STATUS_OFF | 0x0110 | W | 0x00000000 | Correctable Error Status Register. |
| AER_CORR_ERR_MASK_OFF | 0x0114 | W | 0x0000e000 | Correctable Error Mask Register. |
| AER_ADV_ERR_CAP_CTRL_OFF | 0x0118 | W | 0x000000a0 | Advanced Error Capabilities and Control Register. |
| AER_HDR_LOG_0_OFF | 0x011c | W | 0x00000000 | Header Log Register 0. |
| AER_HDR_LOG_1_OFF | 0x0120 | W | 0x00000000 | Header Log Register 1. |
| AER_HDR_LOG_2_OFF | 0x0124 | W | 0x00000000 | Header Log Register 2. |
| AER_HDR_LOG_3_OFF | 0x0128 | W | 0x00000000 | Header Log Register 3. |
| AER_ROOT_ERR_CMD_OF_F | 0x012c | W | 0x00000000 | Root Error Command Register. Exists only in RC mode. |
| AER_ROOT_ERR_STATUS_OFF | 0x0130 | W | 0x48000000 | Root Error Status Register. Exists only in RC mode. |
| AER_ERR_SRC_ID_OFF | 0x0134 | W | 0x00000000 | Error Source Identification Register. Exists only in RC mode. |
| AER_TLP_PREFIX_LOG_1_OFF | 0x0138 | W | 0x00000000 | TLP Prefix Log Register 1. |
| AER_TLP_PREFIX_LOG_2_OFF | 0x013c | W | 0x00000000 | TLP Prefix Log Register 2. |
| AER_TLP_PREFIX_LOG_3_OFF | 0x0140 | W | 0x00000000 | TLP Prefix Log Register 3. |
| AER_TLP_PREFIX_LOG_4_OFF | 0x0144 | W | 0x00000000 | TLP Prefix Log Register 4. |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.17 AER Capability Detail Register Description

AER_AER_EXT_CAP_HDR_OFF

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x148 | NEXT_OFFSET Next Capability Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 19:16 | RO | 0x2 | CAP_VERSION Capability Version. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 15:0 | RO | 0x0001 | CAP_ID AER Extended Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

AER_UNCORR_ERR_STATUS_OFF

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:23 | RO | 0x0 | reserved |
| 22 | RW | 0x0 | INTERNAL_ERR_STATUS Uncorrectable Internal Error Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 21 | RO | 0x0 | reserved |
| 20 | RW | 0x0 | UNSUPPORTED_REQ_ERR_STATUS Unsupported Request Error Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19 | RW | 0x0 | ECRC_ERR_STATUS ECRC Error Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RW | 0x0 | MALF_TLP_ERR_STATUS Malformed TLP Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 17 | RW | 0x0 | REC_OVERFLOW_ERR_STATUS Receiver Overflow Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 16 | RW | 0x0 | UNEXP_CMPLT_ERR_STATUS Unexpected Completion Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15 | RW | 0x0 | CMPLT_ABORT_ERR_STATUS Completer Abort Status. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 14 | RW | 0x0 | CMPLT_TIMEOUT_ERR_STATUS Completion Timeout Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 13 | RW | 0x0 | FC_PROTOCOL_ERR_STATUS Flow Control Protocol Error Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12 | RW | 0x0 | POIS_TLP_ERR_STATUS Poisoned TLP Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | SURPRISE_DOWN_ERR_STATUS Surprise Down Error Status (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RW | 0x0 | DL_PROTOCOL_ERR_STATUS Data Link Protocol Error Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3:0 | RO | 0x0 | reserved |

AER_UNCORR_ERR_MASK_OFF

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | reserved |
| 24 | RO | 0x0 | ATOMIC_EGRESS_BLOCKED_ERR_MASK AtomicOp Egress Block Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23 | RO | 0x0 | reserved |
| 22 | RW | 0x1 | INTERNAL_ERR_MASK Uncorrectable Internal Error Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 21 | RO | 0x0 | reserved |
| 20 | RW | 0x0 | UNSUPPORTED_REQ_ERR_MASK Unsupported Request Error Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19 | RW | 0x0 | ECRC_ERR_MASK ECRC Error Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RW | 0x0 | MALF_TLP_ERR_MASK Malformed TLP Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 17 | RW | 0x0 | REC_OVERFLOW_ERR_MASK Receiver Overflow Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 16 | RW | 0x0 | UNEXP_CMPLT_ERR_MASK Unexpected Completion Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15 | RW | 0x0 | CMPLT_ABORT_ERR_MASK Completer Abort Error Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 14 | RW | 0x0 | CMPLT_TIMEOUT_ERR_MASK Completion Timeout Error Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 13 | RW | 0x0 | FC_PROTOCOL_ERR_MASK Flow Control Protocol Error Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12 | RW | 0x0 | POIS_TLP_ERR_MASK Poisoned TLP Error Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11:6 | RO | 0x0 | reserved |
| 5 | RO | 0x0 | SURPRISE_DOWN_ERR_MASK Surprise Down Error Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RW | 0x0 | DL_PROTOCOL_ERR_MASK Data Link Protocol Error Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3:0 | RO | 0x0 | reserved |

AER UNCORR ERR SEV OFF

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | reserved |
| 24 | RW | 0x0 | ATOMIC_EGRESS_BLOCKED_ERR_SEVERITY AtomicOp Egress Blocked Severity (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23 | RO | 0x0 | reserved |
| 22 | RW | 0x1 | INTERNAL_ERR_SEVERITY Uncorrectable Internal Error Severity (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 21 | RO | 0x0 | reserved |
| 20 | RW | 0x0 | UNSUPPORTED_REQ_ERR_SEVERITY Unsupported Request Error Severity. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19 | RW | 0x0 | ECRC_ERR_SEVERITY ECRC Error Severity (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 18 | RW | 0x1 | MALF_TLP_ERR_SEVERITY Malformed TLP Severity. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 17 | RW | 0x1 | REC_OVERFLOW_ERR_SEVERITY Receiver Overflow Error Severity (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 16 | RW | 0x0 | UNEXP_CMPLT_ERR_SEVERITY Unexpected Completion Error Severity. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15 | RW | 0x0 | CMPLT_ABORT_ERR_SEVERITY Completer Abort Error Severity (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 14 | RW | 0x0 | CMPLT_TIMEOUT_ERR_SEVERITY Completion Timeout Error Severity. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 13 | RW | 0x1 | FC_PROTOCOL_ERR_SEVERITY Flow Control Protocol Error Severity (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12 | RW | 0x0 | POIS_TLP_ERR_SEVERITY Poisoned TLP Severity. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11:6 | RO | 0x0 | reserved |
| 5 | RO | 0x1 | SURPRISE_DOWN_ERR_SVRITY Surprise Down Error Severity (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RO | 0x1 | DL_PROTOCOL_ERR_SEVERITY Data Link Protocol Error Severity. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3:0 | RO | 0x0 | reserved |

AER CORR ERR STATUS OFF

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15 | RW | 0x0 | HEADER_LOG_OVERFLOW_STATUS Header Log Overflow Error Status (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 14 | RW | 0x0 | CORRECTED_INT_ERR_STATUS Corrected Internal Error Status (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 13 | RW | 0x0 | ADVISORY_NON_FATAL_ERR_STATUS Advisory Non-Fatal Error Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12 | RW | 0x0 | RPL_TIMER_TIMEOUT_STATUS Replay Timer Timeout Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | REPLAY_NO_ROLEOVER_STATUS REPLAY_NUM Rollover Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7 | RW | 0x0 | BAD_DLLP_STATUS Bad DLLP Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 6 | RW | 0x0 | BAD_TLP_STATUS Bad TLP Status. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | RX_ERR_STATUS Receiver Error Status (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |

AER CORR ERR MASK OFF

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15 | RW | 0x1 | HEADER_LOG_OVERFLOW_MASK Header Log Overflow Error Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 14 | RW | 0x1 | CORRECTED_INT_ERR_MASK Corrected Internal Error Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |
| 13 | RW | 0x1 | ADVISORY_NON_FATAL_ERR_MASK Advisory Non-Fatal Error Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 12 | RW | 0x0 | RPL_TIMER_TIMEOUT_MASK Replay Timer Timeout Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 11:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | REPLAY_NO_ROLEOVER_MASK REPLAY_NUM Rollover Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7 | RW | 0x0 | BAD_DLLP_MASK Bad DLLP Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 6 | RW | 0x0 | BAD_TLP_MASK Bad TLP Mask. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | RX_ERR_MASK Receiver Error Mask (Optional). For a description of this standard PCIe register field, see the PCI Express Specification. |

AER ADV ERR CAP CTRL OFF

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | reserved |
| 10 | RO | 0x0 | MULTIPLE_HEADER_EN Multiple Header Recording Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 9 | RO | 0x0 | MULTIPLE_HEADER_CAP Multiple Header Recording Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 8 | RW | 0x0 | ECRC_CHECK_EN ECRC Check Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7 | RO | 0x1 | ECRC_CHECK_CAP ECRC Check Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 6 | RW | 0x0 | ECRC_GEN_EN ECRC Generation Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 5 | RO | 0x1 | ECRC_GEN_CAP ECRC Generation Capable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4:0 | RO | 0x00 | FIRST_ERR_POINTER First Error Pointer. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER_HDR_LOG_0_OFF

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | FIRST_DWORD_FOURTH_BYTE Byte 3 of Header log register of First 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | FIRST_DWORD_THIRD_BYTE Byte 2 of Header log register of First 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | FIRST_DWORD_SECOND_BYTE Byte 1 of Header log register of First 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:0 | RO | 0x00 | FIRST_DWORD_FIRST_BYTE Byte 0 of Header log register of First 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER_HDR_LOG_1_OFF

Address: Operational Base + offset (0x0120)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | SECOND_DWORD_FOURTH_BYTE Byte 3 of Header log register of Second 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | SECOND_DWORD_THIRD_BYTE Byte 2 of Header log register of Second 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | SECOND_DWORD_SECOND_BYTE Byte 1 of Header log register of Second 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |
| 7:0 | RO | 0x00 | SECOND_DWORD_FIRST_BYTE Byte 0 of Header log register of Second 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER HDR LOG 2 OFF

Address: Operational Base + offset (0x0124)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | THIRD_DWORD_FOURTH_BYTE Byte 3 of Header log register of Third 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | THIRD_DWORD_THIRD_BYTE Byte 2 of Header log register of Third 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | THIRD_DWORD_SECOND_BYTE Byte 1 of Header log register of Third 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:0 | RO | 0x00 | THIRD_DWORD_FIRST_BYTE Byte 0 of Header log register of Third 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER HDR LOG 3 OFF

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | FOURTH_DWORD_FOURTH_BYTE Byte 3 of Header log register of Fourth 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | FOURTH_DWORD_THIRD_BYTE Byte 2 of Header log register of Fourth 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | FOURTH_DWORD_SECOND_BYTE Byte 1 of Header log register of Fourth 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:0 | RO | 0x00 | FOURTH_DWORD_FIRST_BYTE Byte 0 of Header log register of Fourth 32 bit Data Word. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER_ROOT_ERR_CMD_OFF

Address: Operational Base + offset (0x012c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | FATAL_ERR_REPORTING_EN Fatal Error Reporting Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1 | RW | 0x0 | NON_FATAL_ERR_REPORTING_EN Non-Fatal Error Reporting Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RW | 0x0 | CORR_ERR_REPORTING_EN Correctable Error Reporting Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER_ROOT_ERR_STATUS_OFF

Address: Operational Base + offset (0x0130)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x09 | ADV_ERR_INT_MSG_NUM Advanced Error Interrupt Message Number. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 26:7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | FATAL_ERR_MSG_RX One or more Fatal Error Messages Received. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 5 | RW | 0x0 | NON_FATAL_ERR_MSG_RX One or more Non-Fatal Error Messages Received. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 4 | RW | 0x0 | FIRST_UNCORR_FATAL First Uncorrectable Error is Fatal. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3 | RW | 0x0 | MUL_ERR_FATAL_NON_FATAL_RX Multiple Fatal or Non-Fatal Errors Received. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RW | 0x0 | ERR_FATAL_NON_FATAL_RX Fatal or Non-Fatal Error Received. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1 | RW | 0x0 | MUL_ERR_COR_RX Multiple Correctable Errors Received. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RW | 0x0 | ERR_COR_RX Correctable Error Received. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER_ERR_SRC_ID_OFF

Address: Operational Base + offset (0x0134)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0000 | ERR_FATAL_NON_FATAL_SOURCE_ID Source of Fatal/Non-Fatal Error. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:0 | RO | 0x0000 | ERR_COR_SOURCE_ID Source of Correctable Error. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER_TLP_PREFIX_LOG_1_OFF

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x00 | CFG_TLP_PFX_LOG_1_FOURTH_BYTE Byte 3 of Error TLP Prefix Log 1. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | CFG_TLP_PFX_LOG_1_THIRD_BYTE Byte 2 of Error TLP Prefix Log 1. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | CFG_TLP_PFX_LOG_1_SECOND_BYTE Byte 1 of Error TLP Prefix Log 1. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:0 | RO | 0x00 | CFG_TLP_PFX_LOG_1_FIRST_BYTE Byte 0 of Error TLP Prefix Log 1. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER TLP PREFIX LOG 2 OFF

Address: Operational Base + offset (0x013c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | CFG_TLP_PFX_LOG_2_FOURTH_BYTE Byte 3 Error TLP Prefix Log 2. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | CFG_TLP_PFX_LOG_2_THIRD_BYTE Byte 2 Error TLP Prefix Log 2. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | CFG_TLP_PFX_LOG_2_SECOND_BYTE Byte 1 Error TLP Prefix Log 2. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:0 | RO | 0x00 | CFG_TLP_PFX_LOG_2_FIRST_BYTE Byte 0 Error TLP Prefix Log 2. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER TLP PREFIX LOG 3 OFF

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | WO | 0x00 | CFG_TLP_PFX_LOG_3_FOURTH_BYTE Byte 3 Error TLP Prefix Log 3. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | CFG_TLP_PFX_LOG_3_THIRD_BYTE Byte 2 Error TLP Prefix Log 3. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | CFG_TLP_PFX_LOG_3_SECOND_BYTE Byte 1 Error TLP Prefix Log 3. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:0 | RO | 0x00 | CFG_TLP_PFX_LOG_3_FIRST_BYTE Byte 0 Error TLP Prefix Log 3. For a description of this standard PCIe register field, see the PCI Express Specification. |

AER TLP PREFIX LOG 4 OFF

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x00 | CFG_TLP_PFX_LOG_4_FOURTH_BYTE Byte 3 Error TLP Prefix Log 4. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 23:16 | RO | 0x00 | CFG_TLP_PFX_LOG_4_THIRD_BYTE Byte 2 Error TLP Prefix Log 4. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RO | 0x00 | CFG_TLP_PFX_LOG_4_SECOND_BYTE Byte 1 Error TLP Prefix Log 4. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:0 | RO | 0x00 | CFG_TLP_PFX_LOG_4_FIRST_BYTE Byte 0 Error TLP Prefix Log 4. For a description of this standard PCIe register field, see the PCI Express Specification. |

10.5.18 Secondary PCIe Capability Registers Summary

Secondary PCI Express Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|---|---------------|-------------|--------------------|---|
| SPCIE CAP HEADER REG | 0x0148 | W | 0x15810019 | SPCIE Capability Header. |
| SPCIE LINK CONTROL3 REG | 0x014c | W | 0x00000000 | Link Control 3 Register. |
| SPCIE LANE ERR STATUS REG | 0x0150 | W | 0x00000000 | Lane Error Status Register. |
| SPCIE CAP OFF OCH REG | 0x0154 | W | 0x0f007000 | Lane Equalization Control Register for lanes 1 and 0. |

Notes: **S**-Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access**10.5.19 Secondary PCIe Capability Detail Register Description****SPCIE CAP HEADER REG**

Address: Operational Base + offset (0x0148)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x158 | <p>NEXT_OFFSET Next Capability Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R Note: the reset value is 0x160 in RC mode.</p> |
| 19:16 | RO | 0x1 | <p>CAP_VERSION Capability Version. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R</p> |
| 15:0 | RO | 0x0019 | <p>EXTENDED_CAP_ID Secondary PCI Express Extended Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R</p> |

SPCIE LINK CONTROL3 REG

Address: Operational Base + offset (0x014c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1 | RO | 0x0 | <p>EQ_REQ_INT_EN Link Equalization Request Interrupt Enable. For a description of this standard PCIe register field, see the PCI Express Specification.</p> |
| 0 | RO | 0x0 | <p>PERFORM_EQ Perform Equalization. For a description of this standard PCIe register field, see the PCI Express Specification.</p> |

SPCIE LANE ERR STATUS REG

Address: Operational Base + offset (0x0150)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x0 | <p>LANE_ERR_STATUS Lane Error Status Bits per Lane. For a description of this standard PCIe register field, see the PCI Express Specification.</p> |

SPCIE CAP OFF OCH REG

Address: Operational Base + offset (0x0154)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RO | 0x0 | reserved |
| 30:28 | RO | 0x0 | USP_RX_PRESET_HINT1 Upstream Port 8.0 GT/s Receiver Preset Hint 1. |
| 27:24 | RO | 0xf | USP_TX_PRESET1 Upstream Port 8.0 GT/s Transmitter Preset 1. |
| 23 | RO | 0x0 | reserved |
| 22:20 | RO | 0x0 | DSP_RX_PRESET_HINT1 Downstream Port 8.0 GT/s Receiver Preset Hint 1. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19:16 | RO | 0x0 | DSP_TX_PRESET1 Downstream Port 8.0 GT/s Transmitter Preset 1. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15 | RO | 0x0 | reserved |
| 14:12 | RO | 0x7 | USP_RX_PRESET_HINT0 Upstream Port 8.0 GT/s Receiver Preset Hint 0. |
| 11:8 | RO | 0x0 | USP_TX_PRESET0 Upstream Port 8.0 GT/s Transmitter Preset 0. |
| 7 | RO | 0x0 | reserved |
| 6:4 | RO | 0x0 | DSP_RX_PRESET_HINT0 Downstream Port 8.0 GT/s Receiver Preset Hint 0. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3:0 | RO | 0x0 | DSP_TX_PRESET0 Downstream Port 8.0 GT/s Transmitter Preset 0. For a description of this standard PCIe register field, see the PCI Express Specification. |

10.5.20 LTR Capability Registers Summary

PF Latency Tolerance Reporting Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|-----------------|---------------|-------------|--------------------|--|
| LTR_CAP_HDR_REG | 0x0158 | W | 0x16010018 | LTR Extended Capability Header. Exists only in EP mode. |
| LTR_LATENCY_REG | 0x015c | W | 0x00000000 | LTR Max Snoop and No-Snoop Latency Register. Exists only in EP mode. |

Notes: **S**-Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.21 LTR Capability Detail Register Description

LTR_CAP_HDR_REG

Address: Operational Base + offset (0x0158)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:20 | RO | 0x160 | NEXT_OFFSET Next Capability Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 19:16 | RO | 0x1 | CAP_VERSION Capability Version. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 15:0 | RO | 0x0018 | CAP_ID LTR Extended Capacity ID. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

LTR LATENCY REG

Address: Operational Base + offset (0x015c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:29 | RO | 0x0 | reserved |
| 28:26 | RW | 0x0 | MAX_NO_SNOOP_LAT_SCALE Max No-Snoop Latency Scale. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 25:16 | RW | 0x000 | MAX_NO_SNOOP_LAT Max No-Snoop Latency Value. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:13 | RO | 0x0 | reserved |
| 12:10 | RW | 0x0 | MAX_SNOOP_LAT_SCALE Max Snoop Latency Scale. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 9:0 | RW | 0x000 | MAX_SNOOP_LAT Max Snoop Latency Value. For a description of this standard PCIe register field, see the PCI Express Specification. |

10.5.22 L1 Substates Capability Registers Summary

L1 Substates Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|----------------------|--------|------|-------------|--|
| L1SUB CAP HEADER REG | 0x0160 | W | 0x1701001e | L1 Substates Extended Capability Header. |
| L1SUB CAPABILITY REG | 0x0164 | W | 0x00280a1f | L1 Substates Capability Register. |
| L1SUB CONTROL1 REG | 0x0168 | W | 0x00000000 | L1 Substates Control 1 Register. |
| L1SUB CONTROL2 REG | 0x016c | W | 0x00000028 | L1 Substates Control 2 Register. |

Notes: **S**-Size: **B**- Byte (8 bits) access, **H**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.23 L1 Substates Capability Detail Register Description**L1SUB CAP HEADER REG**

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x170 | NEXT_OFFSET Next Capability Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: the reset value is 0x1a0 in RC mode. |
| 19:16 | RO | 0x1 | CAP_VERSION Capability Version. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |
| 15:0 | RO | 0x001e | EXTENDED_CAP_ID L1SUB Extended Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) |

L1SUB CAPABILITY REG

Address: Operational Base + offset (0x0164)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23:19 | RW | 0x05 | PWR_ON_VALUE_SUPPORT Port T Power On Value. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 18 | RO | 0x0 | reserved |
| 17:16 | RW | 0x0 | PWR_ON_SCALE_SUPPORT Port T Power On Scale. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RW | 0x0a | COMM_MODE_SUPPORT Port Common Mode Restore Time. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x1 | L1_PMSUB_SUPPORT L1 PM Substates ECN Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 3 | RW | 0x1 | L1_1_ASPM_SUPPORT ASPM L11 Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RW | 0x1 | L1_2_ASPM_SUPPORT ASPM L12 Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 1 | RW | 0x1 | L1_1_PCIPM_SUPPORT PCI-PM L11 Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RW | 0x1 | L1_2_PCIPM_SUPPORT PCI-PM L12 Supported. For a description of this standard PCIe register field, see the PCI Express Specification. |

L1SUB CONTROL1 REG

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RW | 0x0 | L1_2_TH_SCA LTR L12 Threshold Scale. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 28:26 | RO | 0x0 | reserved |
| 25:16 | RW | 0x000 | L1_2_TH_VAL LTR L12 Threshold Value. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:8 | RW | 0x00 | T_COMMON_MODE Common Mode Restore Time. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 7:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | L1_1_ASPM_EN ASPM L11 Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RW | 0x0 | L1_2_ASPM_EN ASPM L12 Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 1 | RW | 0x0 | L1_1_PCIPM_EN PCI-PM L11 Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 0 | RW | 0x0 | L1_2_PCIPM_EN PCI-PM L12 Enable. For a description of this standard PCIe register field, see the PCI Express Specification. |

L1SUB CONTROL2 REG

Address: Operational Base + offset (0x016c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:3 | RW | 0x05 | T_POWER_ON_VALUE T Power On Value. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x0 | T_POWER_ON_SCALE T Power On Scale. For a description of this standard PCIe register field, see the PCI Express Specification. |

10.5.24 DPA Capability Registers Summary

PF DPA Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|-----------------------|--------|------|-------------|--|
| DPA EXT CAP HDR REG | 0x0170 | W | 0x1a010016 | DPA Extended Capability Header. Exists only in EP mode. |
| DPA CAP REG | 0x0174 | W | 0xff801107 | DPA Capability Register. Exists only in EP mode. |
| DPA LAT IND REG | 0x0178 | W | 0x00000000 | DPA Latency Indicator Register. Exists only in EP mode. |
| DPA STATUS CNTRL REG | 0x017c | W | 0x000000100 | DPA Status and Control Register. Exists only in EP mode. |
| DPA PWR ALLOC ARRAY 0 | 0x0180 | W | 0x1d1e1f20 | DSP Substate Power Allocation Register 3..0. Exists only in EP mode. |
| DPA PWR ALLOC ARRAY 4 | 0x0184 | W | 0x191a1b1c | DSP Substate Power Allocation Register 7..4. Exists only in EP mode. |

Notes: **S**-ize: **B**- Byte (8 bits) access, **H**W- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.25 DPA Capability Detail Register Description

DPA EXT CAP HDR REG

Address: Operational Base + offset (0x0170)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:20 | RO | 0x1a0 | NEXT_OFFSET Next Capability Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |
| 19:16 | RO | 0x1 | CAP_VERSION Capability Version. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 15:0 | RO | 0x0016 | <p>EXT_CAP_ID DPA Extended Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W else R</p> |

DPA CAP REG

Address: Operational Base + offset (0x0174)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0xff | <p>XLCY1 Transition Latency Value 1 (Xlcy1). For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky)</p> |
| 23:16 | RW | 0x80 | <p>XLCY0 Transition Latency Value 0 (Xlcy0). For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky)</p> |
| 15:14 | RO | 0x0 | reserved |
| 13:12 | RW | 0x1 | <p>PAS Power Allocation Scale. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky)</p> |
| 11:10 | RO | 0x0 | reserved |
| 9:8 | RW | 0x1 | <p>TLUNIT Transition Latency Unit (TLunit). For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky)</p> |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x07 | <p>SUBSTATE_MAX Maximum Substate Number. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky)</p> |

DPA LAT IND REG

Address: Operational Base + offset (0x0178)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | X1_INDICATOR1 Transition Latency Indicator Bits. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |

DPA STATUS CNTRL REG

Address: Operational Base + offset (0x017c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | SUBSTATE_CONTROL Substate Control. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:9 | RO | 0x0 | reserved |
| 8 | RW | 0x1 | SUBSTATE_CONTROL_EN Substate Control Enabled. For a description of this standard PCIe register field, see the PCI Express Specification. Note:: RW for DBI Access Note: The access attributes of this field are as follows: Wire: RW1C Dbi: R/W |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RW | 0x00 | SUBSTATE_STATUS Substate Status. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R Dbi: R/W |

DPA PWR ALLOC ARRAY

Address: Operational Base + offset (0x0180)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x1d | PWR_ALLOC_VAL3 Substate Power Allocation Substate 3. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 23:16 | RW | 0x1e | PWR_ALLOC_VAL2 Substate Power Allocation Substate 2. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |
| 15:8 | RW | 0x1f | PWR_ALLOC_VAL1 Substate Power Allocation Substate 1. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |
| 7:0 | RW | 0x20 | PWR_ALLOC_VAL0 Substate Power Allocation Substate 0. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |

DPA PWR ALLOC ARRAY4

Address: Operational Base + offset (0x0184)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x19 | PWR_ALLOC_VAL7 Substate Power Allocation Substate 7. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |
| 23:16 | RW | 0x1a | PWR_ALLOC_VAL6 Substate Power Allocation Substate 6. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |
| 15:8 | RW | 0x1b | PWR_ALLOC_VAL5 Substate Power Allocation Substate 5. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 7:0 | RW | 0x1c | PWR_ALLOC_VAL4 Substate Power Allocation Substate 4. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: R/W (sticky) |

10.5.26 RAS DES Capability Registers Summary

RAS D.E.S. Capability Structure (VSEC) Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|--|--------|------|-------------|--|
| RAS DES CAP HEADER REG | 0x01a0 | W | 0x2a01000b | Vendor-Specific Extended Capability Header. |
| RAS DES VENDOR SPECIFIC HEADER REG | 0x01a4 | W | 0x10040002 | Vendor-Specific Header. |
| RAS DES EINJ ENABLE REG | 0x01d0 | W | 0x00000000 | Error Injection Enable. Each type of error insertion is enabled by the corresponding bit in this register. |
| RAS DES EINJ0_CRC_REG | 0x01d4 | W | 0x00000000 | Error Injection Control 0 (CRC Error). Controls the insertion of errors into the CRC, and parity of ordered sets for the selected type of the packets. |
| RAS DES EINJ1_SEQNUM_REG | 0x01d8 | W | 0x00000000 | Error Injection Control 1 (Sequence Number Error). Controls the sequence number of the specific TLPs and ACK/NAK DLLPs. |
| RAS DES EINJ2_DLLP_REG | 0x01dc | W | 0x00000000 | Error Injection Control 2 (DLLP Error). Controls the transmission of DLLPs and inserts errors. |
| RAS DES EINJ3_SYMBOL_REG | 0x01e0 | W | 0x00000000 | Error Injection Control 3 (Symbol Error). When 8b/10b encoding is used, this register controls error insertion into the special (K code) symbols. |
| RAS DES EINJ4_FC_REG | 0x01e4 | W | 0x00000000 | Error Injection Control 4 (FC Credit Error). Controls error insertion into the credit value in the UpdateFCs. |

| Name | Offset | Size | Reset Value | Description |
|------------------------------------|--------|------|-------------|--|
| RAS DES EINJ5 SP TLP REG | 0x01e8 | W | 0x00000000 | Error Injection Control 5 (Specific TLP Error). Controls the generation of specified TLPs. Correctable errors will occur which will be fixed by the PCIe protocol. |
| RAS DES EINJ6 COMPARE POINT H0 REG | 0x01ec | W | 0x00000000 | Error Injection Control 6 (Compare Point Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. |
| RAS DES EINJ6 COMPARE POINT H1 REG | 0x01f0 | W | 0x00000000 | Error Injection Control 6 (Compare Point Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. |
| RAS DES EINJ6 COMPARE POINT H2 REG | 0x01f4 | W | 0x00000000 | Error Injection Control 6 (Compare Point Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. |
| RAS DES EINJ6 COMPARE POINT H3 REG | 0x01f8 | W | 0x00000000 | Error Injection Control 6 (Compare Point Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. |
| RAS DES EINJ6 COMPARE VALUE H0 REG | 0x01fc | W | 0x00000000 | Error Injection Control 6 (Compare Value Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. |
| RAS DES EINJ6 COMPARE VALUE H1 REG | 0x0200 | W | 0x00000000 | Error Injection Control 6 (Compare Value Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. |
| RAS DES EINJ6 COMPARE VALUE H2 REG | 0x0204 | W | 0x00000000 | Error Injection Control 6 (Compare Value Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. |
| RAS DES EINJ6 COMPARE VALUE H3 REG | 0x0208 | W | 0x00000000 | Error Injection Control 6 (Compare Value Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. |
| RAS DES EINJ6 CHANGE POINT H0 REG | 0x020c | W | 0x00000000 | Error Injection Control 6 (Change Point Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. |

| Name | Offset | Size | Reset Value | Description |
|-----------------------------------|--------|------|-------------|---|
| RAS DES EINJ6 CHANGE POINT H1 REG | 0x0210 | W | 0x00000000 | Error Injection Control 6 (Change Point Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. |
| RAS DES EINJ6 CHANGE POINT H2 REG | 0x0214 | W | 0x00000000 | Error Injection Control 6 (Change Point Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. |
| RAS DES EINJ6 CHANGE POINT H3 REG | 0x0218 | W | 0x00000000 | Error Injection Control 6 (Change Point Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. |
| RAS DES EINJ6 CHANGE VALUE H0 REG | 0x021c | W | 0x00000000 | Error Injection Control 6 (Change Value Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. |
| RAS DES EINJ6 CHANGE VALUE H1 REG | 0x0220 | W | 0x00000000 | Error Injection Control 6 (Change Value Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. |
| RAS DES EINJ6 CHANGE VALUE H2 REG | 0x0224 | W | 0x00000000 | Error Injection Control 6 (Change Value Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. |
| RAS DES EINJ6 CHANGE VALUE H3 REG | 0x0228 | W | 0x00000000 | Error Injection Control 6 (Change Value Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. |
| RAS DES EINJ6 TLP REG | 0x022c | W | 0x00000000 | Error Injection Control 6 (Packet Error). |
| RAS DES SD CONTROL1 REG | 0x0240 | W | 0x00000000 | Silicon Debug Control 1. |
| RAS DES SD CONTROL2 REG | 0x0244 | W | 0x00000000 | Silicon Debug Control 2. |
| RAS DES SD STATUS L1 LANE REG | 0x0250 | W | 0x00180000 | Silicon Debug Status(Layer1 Per-lane). |
| RAS DES SD STATUS L1 LTSSM REG | 0x0254 | W | 0x00000200 | Silicon Debug Status(Layer1 LTSSM). |
| RAS DES SD STATUS PM REG | 0x0258 | W | 0x00000000 | Silicon Debug Status(PM). |
| RAS DES SD STATUS L2 REG | 0x025c | W | 0x00fff000 | Silicon Debug Status(Layer2). |
| RAS DES SD STATUS L3 FC REG | 0x0260 | W | 0x00000000 | Silicon Debug Status(Layer3 FC). |

| Name | Offset | Size | Reset Value | Description |
|-----------------------------|--------|------|-------------|-------------------------------|
| RAS DES SD STATUS L3 REG | 0x0264 | W | 0x00000000 | Silicon Debug Status(Layer3). |
| RAS DES SD EQ CONTR OL1 REG | 0x0270 | W | 0x00000000 | Silicon Debug EQ Control 1. |
| RAS DES SD EQ CONTR OL2 REG | 0x0274 | W | 0x00000000 | Silicon Debug EQ Control 2. |
| RAS DES SD EQ CONTR OL3 REG | 0x0278 | W | 0x00000000 | Silicon Debug EQ Control 3. |
| RAS DES SD EQ STATUS1 REG | 0x0280 | W | 0x00000000 | Silicon Debug EQ Status 1. |
| RAS DES SD EQ STATUS2 REG | 0x0284 | W | 0x00000000 | Silicon Debug EQ Status 2. |
| RAS DES SD EQ STATUS3 REG | 0x0288 | W | 0x00000000 | Silicon Debug EQ Status 3. |

Notes:Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.27 RAS DES Capability Detail Register Description

RAS DES CAP HEADER REG

Address: Operational Base + offset (0x01a0)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:20 | RO | 0x2a0 | <p>NEXT_OFFSET Next Capability Offset. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: The reset value is 0x0 in RC mode.</p> |
| 19:16 | RO | 0x1 | <p>CAP_VERSION Capability Version. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |
| 15:0 | RO | 0x000b | <p>EXTENDED_CAP_ID PCI Express Extended Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> |

RAS DES VENDOR SPECIFIC HEADER REG

Address: Operational Base + offset (0x01a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x100 | VSEC_LENGTH VSEC Length. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 19:16 | RO | 0x4 | VSEC_REV VSEC Rev. For a description of this standard PCIe register field, see the PCI Express Specification. |
| 15:0 | RO | 0x0002 | VSEC_ID VSEC ID. For a description of this standard PCIe register field, see the PCI Express Specification. |

RAS DES EINJ ENABLE REG

Address: Operational Base + offset (0x01d0)

Description: Error Injection Enable. Each type of error insertion is enabled by the corresponding bit in this register. The specific injection controls for each type of error are defined in the following registers:

- 0: CRC Error: EINJ0_CRC_REG
- 1: Sequence Number Error: EINJ1_SEQNUM_REG
- 2: DLLP Error: EINJ2_DLLP_REG
- 3: Symbol DataK Mask Error or Sync Header Error: EINJ3_SYMBOL_REG
- 4: FC Credit Update Error: EINJ4_FC_REG
- 5: TLP Duplicate/Nullify Error: EINJ5_SP_TLP_REG
- 6: Specific TLP Error:
EINJ6_COMPARE_*_REG/EINJ6_CHANGE_*_REG/EINJ6_TLP_REG

After the errors have been inserted by controller, it will clear each bit here.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:7 | RO | 0x0 | reserved |
| 6 | RW | 0x0 | ERROR_INJECTION6_ENABLE Error Injection6 Enable (Specific TLP Error). Enables insertion of errors into the packets that you select. You can set this bit to '1' when you have disabled RAS datapath protection (DP) by setting CX_RASDP = CX_RASDP_RAM_PROT =0. You can set this bit to '1' when you have disabled the address translation by setting ADDR_TRANSLATION_SUPPORT_EN=0. For more details, see the EINJ6_COMPARE_*_REG/EINJ6_CHANGE_*_REG/EINJ6_TLP_REG registers. Note: This register field is sticky. |
| 5 | RW | 0x0 | ERROR_INJECTION5_ENABLE Error Injection5 Enable (TLP Duplicate/Nullify Error). Enables insertion of duplicate/nullified TLPs. For more details, see the EINJ5_SP_TLP_REG register. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 4 | RW | 0x0 | <p>ERROR_INJECTION4_ENABLE Error Injection4 Enable (FC Credit Update Error). Enables insertion of errors into UpdateFCs. For more details, see the EINJ4_FC_REG register. Note: This register field is sticky.</p> |
| 3 | RW | 0x0 | <p>ERROR_INJECTION3_ENABLE Error Injection3 Enable (Symbol DataK Mask Error or Sync Header Error). Enables DataK masking of special symbols or the breaking of the sync header. For more details, see the EINJ3_SYMBOL_REG register. Note: This register field is sticky.</p> |
| 2 | RW | 0x0 | <p>ERROR_INJECTION2_ENABLE Error Injection2 Enable (DLLP Error). Enables insertion of DLLP errors. For more details, see the EINJ2_DLLP_REG register. Note: This register field is sticky.</p> |
| 1 | RW | 0x0 | <p>ERROR_INJECTION1_ENABLE Error Injection1 Enable (Sequence Number Error). Enables insertion of errors into sequence numbers. For more details, see the EINJ1_SEQNUM_REG register. Note: This register field is sticky.</p> |
| 0 | RW | 0x0 | <p>ERROR_INJECTION0_ENABLE Error Injection0 Enable (CRC Error). Enables insertion of errors into various CRC. For more details, see the EINJ0_CRC_REG register. Note: This register field is sticky.</p> |

RAS DES EINJ0 CRC REG

Address: Operational Base + offset (0x01d4)

Description: Error Injection Control 0 (CRC Error). Controls the insertion of errors into the CRC, and parity of ordered sets for the selected type of the packets as follows:

- LCRC. Bad TLP will be detected at the receiver side; receiver responds with NAK DLLP; Data Link Retry starts.
- 16-bit CRC of ACK/NAK DLLPs. Bad DLLP occurs at the receiver side; Replay NUM Rollover occurs.
- 16-bit CRC of UpdateFC DLLPs. Error insertion continues for the specific time; LTSSM transitions to the Recovery state because of the UpdateFC timeout (if the timeout is implemented at the receiver of the UpdateFCs).
- ECRC. If ECRC check is enabled, ECRC error is detected at the receiver side.
- FCRC. Framing error will be detected, TLP is discarded, and the LTSSM transitions to Recovery state.
- Parity of TSOS. Error insertion continues for the specific time; LTSSM Recovery/Configuration timeout will occur.
- Parity of SKPOS. Lane error will be detected at the receiver side.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:12 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11:8 | RW | 0x0 | <p>EINJ0_CRC_TYPE Error injection type. Selects the type of CRC error to be inserted. Tx Path 0000b: New TLP's LCRC error injection 0001b: 16bCRC error injection of ACK/NAK DLLP 0010b: 16bCRC error injection of Update-FC DLLP 0011b: New TLP's ECRC error injection 0100b: TLP's FCRC error injection (128b/130b) 0101b: Parity error of TSOS (128b/130b) 0110b: Parity error of SKPOS (128b/130b) Rx Path 1000b: LCRC error injection 1011b: ECRC error injection Others: Reserved Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>EINJ0_COUNT Error injection count. Indicates the number of errors. This register is decremented when the errors have been inserted. If the counter value is 0x01 and error is inserted, ERROR_INJECTION0_ENABLE in EINJ_ENABLE_REG returns 0b. If the counter value is 0x00 and ERROR_INJECTION0_ENABLE=1, the errors are inserted until ERROR_INJECTION0_ENABLE is set to '0'. Note: This register field is sticky.</p> |

RAS DES EINJ1 SEQNUM REG

Address: Operational Base + offset (0x01d8)

Description: Error Injection Control 1 (Sequence Number Error).

Controls the sequence number of the specific TLPs and ACK/NAK DLLPs. Data Link Protocol Error will be detected at the Rx side of ACK/NAL DLLPs when one of these conditions is true:

- $((\text{NEXT_TRANSMIT_SEQ} - 1) - \text{AckNak_Seq_Num}) \bmod 4096 > 2048$
- $(\text{AckNak_Seq_Num} - \text{ACKD_SEQ}) \bmod 4096 \geq 2048$

TLP is treated as Duplicate TLP at the Rx side when all these conditions are true:

- Sequence Number $\neq \text{NEXT_RCV_SEQ}$
- $(\text{NEXT_RCV_SEQ} - \text{Sequence Number}) \bmod 4096 \leq 2048$

TLP is treated as Bad TLP at the Rx side when all these conditions are true:

- Sequence Number $\neq \text{NEXT_RCV_SEQ}$ and
- $(\text{NEXT_RCV_SEQ} - \text{Sequence Number}) \bmod 4096 > 2048$

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:29 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 28:16 | RW | 0x0000 | <p>EINJ1_BAD_SEQNUM Bad sequence number. Indicates the value to add/subtract from the naturally-assigned sequence numbers. This value is represented by two's complement.</p> <p>0xFFFF: +4095 .. 0x0002: +2 0x0001: +1 0x0000: 0 0x1FFF: -1 0x1FFE: -2 .. 0x1001: -4095</p> <p>For example: Set Type, SEQ# and Count EINJ1_SEQNUM_TYPE =0 (Insert errors into new TLPs) EINJ1_BAD_SEQNUM =0x1FFD (represents -3) EINJ1_COUNT =1 Enable Error Injection ERROR_INJECTION1_ENABLE =1 Send a TLP From the Core's Application Interface Assume SEQ#5 is given to the TLP. The SEQ# is Changed to #2 by the Error Injection Function in Layer2. $5 + (-3) = 2$ The TLP with SEQ#2 is Transmitted to PCIe Link. Note: This register field is sticky.</p> |
| 15:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | <p>EINJ1_SEQNUM_TYPE Sequence number type. Selects the type of sequence number. 0b: Insertion of New TLP's SEQ# error 1b: Insertion of ACK/NAK DLLP's SEQ# Error</p> |
| 7:0 | RW | 0x00 | <p>EINJ1_COUNT Error injection count. Indicates the number of errors. This register is decremented as the errors are being inserted. If the counter value is 0x01 and error is inserted, ERROR_INJECTION1_ENABLE in EINJ_ENABLE_REG returns '0'. If the counter value is 0x00 and ERROR_INJECTION1_ENABLE=1, the errors are inserted until ERROR_INJECTION1_ENABLE is set to '0'. Note: This register field is sticky.</p> |

RAS DES EINJ2 DLLP REG

Address: Operational Base + offset (0x01dc)

Description: Error Injection Control 2 (DLLP Error). Controls the transmission of DLLPs and inserts the following errors:

- If "ACK/NAK DLLP's transmission block" is selected, replay timeout error will occur at the transmitter of the TLPs and then Data Link Retry will occur.
- If "Update FC DLLP's transmission block" is selected, LTSSM will transition to the Recovery state because of the UpdateFC timeout (if the timeout is implemented at the receiver of the UpdateFCs).
- If "Always Transmission for NAK DLLP" is selected, Data Link Retry will occur at the transmitter of the TLPs. Furthermore, Replay NUM Rollover will occur when the transmitter has been requested four times to send the TLP with the same sequence number.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | reserved |
| 9:8 | RW | 0x0 | <p>EINJ2_DLLP_TYPE DLLP Type. Selects the type of DLLP errors to be inserted. 00b: ACK/NAK DLLP's transmission block 01b: Update FC DLLP's transmission block 10b: Always Transmission for NAK DLLP 11b: Reserved</p> <p>Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>EINJ2_COUNT Error injection count. Indicates the number of errors. This register is decremented as the errors are being inserted. If the counter value is 0x01 and the error is inserted, ERROR_INJECTION2_ENABLE in EINJ_ENABLE_REG returns '0'. If the counter value is 0x00 and ERROR_INJECTION2_ENABLE =1, the errors are inserted until ERROR_INJECTION2_ENABLE is set to '0'. This register is affected only when EINJ2_DLLP_TYPE =2'10b.</p> <p>Note: This register field is sticky.</p> |

RAS DES EINJ3 SYMBOL REG

Address: Operational Base + offset (0x01e0)

Description: Error Injection Control 3 (Symbol Error). When 8b/10b encoding is used, this register controls error insertion into the special (K code) symbols.

- If TS1/TS2/FTS/E-Idle/SKP is selected, it affects whole of the ordered set. It might cause timeout of the LTSSM.
- If END/EDB/STP/SDP is selected, TLP/DLLP will be discarded at the receiver side.

When 128b/130b encoding is used, this register controls error insertion into the sync-header.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:11 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 10:8 | RW | 0x0 | <p>EINJ3_SYMBOL_TYPE Error Type. 8b/10b encoding - Mask K symbol. It is not supported to insert errors into the first ordered-set after exiting from TxElecIdle when CX_FREQ_STEP_EN has been enabled.</p> <p>000b: Reserved 001b: COM/PAD(TS1 Order set) 010b: COM/PAD(TS2 Order set) 011b: COM/FTS(FTS Order set) 100b: COM/IDL(E-Idle Order set) 101b: END/EDB Symbol 110b: STP/SDP Symbol 111b: COM/SKP(SKP Order set) 128b/130b encoding - Change sync header. 000b: Invert sync header Others: Reserved Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>EINJ3_COUNT Error injection count. Indicates the number of errors. This register is decremented as the errors are being inserted. If the counter value is 0x01 and error is inserted, ERROR_INJECTION3_ENABLE in EINJ_ENABLE_REG returns '0'. If the counter value is 0x00 and ERROR_INJECTION3_ENABLE =1, the errors are inserted until ERROR_INJECTION3_ENABLE is set to '0'. Note: This register field is sticky.</p> |

RAS DES EINJ4 FC REG

Address: Operational Base + offset (0x01e4)

Description: Error Injection Control 4 (FC Credit Error). Controls error insertion into the credit value in the UpdateFCs. It is possible to insert errors for any of the following types:

- o Posted TLP Header credit
- o Non-Posted TLP Header credit
- o Completion TLP Header credit
- o Posted TLP Data credit
- o Non-Posted TLP Data credit
- o Completion TLP Data credit

These errors are not correctable while error insertion is enabled. Receiver buffer overflow error might occur.

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|-------------|
| 31:29 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 28:16 | RW | 0x0000 | <p>EINJ4_BAD_UPDFC_VALUE Bad update-FC credit value. Indicates the value to add/subtract from the UpdateFC credit. This value is represented by two's complement.</p> <p>0x0FFF: +4095 .. 0x0002: +2 0x0001: +1 0x0000: 0 0x1FFF: -1 0x1FFE: -2 .. 0x1001: -4095</p> <p>Note: This register field is sticky.</p> |
| 15 | RO | 0x0 | reserved |
| 14:12 | RW | 0x0 | <p>EINJ4_VC_NUMBER VC Number. Indicates target VC Number.</p> <p>Note: This register field is sticky.</p> |
| 11 | RO | 0x0 | reserved |
| 10:8 | RW | 0x0 | <p>EINJ4_UPDFC_TYPE Update-FC type. Selects the credit type.</p> <p>000b: Posted TLP Header Credit value control 001b: Non-Posted TLP Header Credit value control 010b: Completion TLP Header Credit value control 011b: Reserved 100b: Posted TLP Data Credit value control 101b: Non-Posted TLP Data Credit value control 110b: Completion TLP Data Credit value control 111b: Reserved</p> <p>Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>EINJ4_COUNT Error injection count. Indicates the number of errors. This register is decremented as the errors are being inserted.</p> <p>If the counter value is 0x01 and error is inserted, ERROR_INJECTION4_ENABLE in EINJ_ENABLE_REG returns '0'.</p> <p>If the counter value is 0x00 and ERROR_INJECTION4_ENABLE =1, the errors are inserted until ERROR_INJECTION4_ENABLE is set to '0'.</p> <p>Note: This register field is sticky.</p> |

RAS DES EINJ5 SP TLP REG

Address: Operational Base + offset (0x01e8)

Description: Error Injection Control 5 (Specific TLP Error). Controls the generation of specified TLPs. Correctable errors will occur which will be fixed by the PCIe protocol.

- For Duplicate TLP, the controller initiates Data Link Retry by handling ACK DLLP as NAK DLLP. These TLPs will be duplicate TLPs at the receiver side.

- For Nullified TLP, the TLPs that the controller transmits are changed into nullified TLPs and the original TLPs are stored in the retry buffer. The receiver of these TLPs will detect the lack of seq# and send NAK DLLP at the next TLP. Then the original TLPs are sent from retry buffer and the data controls are recovered. For 128 bit controller or more than 128 bit, the controller inserts errors the number of times of EINJ5_COUNT but doesn't ensure that the errors are continuously inserted into TLPs.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | <p>EINJ5_SPECIFIED_TLP Specified TLP. Selects the specified TLP to be inserted. 0: Generates duplicate TLPs by handling ACK DLLP as NAK DLLP. 1: Generates Nullified TLP (Original TLP will be stored in retry buffer).</p> <p>Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>EINJ5_COUNT Error injection count. Indicates the number of errors. This register is decremented as the errors are being inserted. If the counter value is 0x01 and error is inserted, ERROR_INJECTION5_ENABLE in EINJ_ENABLE_REG returns '0'. If the counter value is 0x00 and ERROR_INJECTION5_ENABLE =1, the errors are inserted until ERROR_INJECTION5_ENABLE is set to '0'. Note: This register field is sticky.</p> |

RAS DES EINJ6 COMPARE POINT H0 REG

Address: Operational Base + offset (0x01ec)

Description: Error Injection Control 6 (Compare Point Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW0[7:0], TLP_DW0[15:8], TLP_DW0[23:16], TLP_DW0[31:24]

The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>EINJ6_COMPARE_POINT_H0 Packet Compare Point: 1st DWORD. Specifies which Tx TLP header DWORD#0 bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. Note: This register field is sticky.</p> |

RAS DES EINJ6 COMPARE POINT H1 REG

Address: Operational Base + offset (0x01f0)

Description: Error Injection Control 6 (Compare Point Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW1[7:0], TLP_DW1[15:8], TLP_DW1[23:16], TLP_DW1[31:24]

The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>EINJ6_COMPARE_POINT_H1 Packet Compare Point: 2nd DWORD. Specifies which Tx TLP header DWORD#1 bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. Note: This register field is sticky.</p> |

RAS DES EINJ6_COMPARE_POINT_H2 REG

Address: Operational Base + offset (0x01f4)

Description: Error Injection Control 6 (Compare Point Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW2[7:0], TLP_DW2[15:8], TLP_DW2[23:16], TLP_DW2[31:24]

The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>EINJ6_COMPARE_POINT_H2 Packet Compare Point: 3rd DWORD. Specifies which Tx TLP header DWORD#2 bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. Note: This register field is sticky.</p> |

RAS DES EINJ6_COMPARE_POINT_H3 REG

Address: Operational Base + offset (0x01f8)

Description: Error Injection Control 6 (Compare Point Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW3[7:0], TLP_DW3[15:8], TLP_DW3[23:16], TLP_DW3[31:24]

The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and

EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>EINJ6_COMPARE_POINT_H3 Packet Compare Point: 4th DWORD. Specifies which Tx TLP header DWORD#3 bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*).</p> <p>When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP.</p> <p>Note: This register field is sticky.</p> |

RAS DES EINJ6 COMPARE VALUE H0 REG

Address: Operational Base + offset (0x01fc)

Description: Error Injection Control 6 (Compare Value Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW0[7:0], TLP_DW0[15:8], TLP_DW0[23:16], TLP_DW0[31:24]
The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>EINJ6_COMPARE_VALUE_H0 Packet Compare Value: 1st DWORD. Specifies the value to compare against Tx the TLP header DWORD#0 bits specified in the Packet Compare Point registers (EINJ6_COMPARE_POINT*).</p> <p>Note: This register field is sticky.</p> |

RAS DES EINJ6 COMPARE VALUE H1 REG

Address: Operational Base + offset (0x0200)

Description: Error Injection Control 6 (Compare Value Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW1[7:0], TLP_DW1[15:8], TLP_DW1[23:16], TLP_DW1[31:24]
The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EINJ6_COMPARE_VALUE_H1 Packet Compare Value: 2nd DWORD. Specifies the value to compare against Tx the TLP header DWORD#1 bits specified in the Packet Compare Point registers (EINJ6_COMPARE_POINT*). Note: This register field is sticky. |

RAS DES EINJ6 COMPARE VALUE H2 REG

Address: Operational Base + offset (0x0204)

Description: Error Injection Control 6 (Compare Value Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW2[7:0], TLP_DW2[15:8], TLP_DW2[23:16], TLP_DW2[31:24]
The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EINJ6_COMPARE_VALUE_H2 Packet Compare Value: 3rd DWORD. Specifies the value to compare against Tx the TLP header DWORD#2 bits specified in the Packet Compare Point registers (EINJ6_COMPARE_POINT*). Note: This register field is sticky. |

RAS DES EINJ6 COMPARE VALUE H3 REG

Address: Operational Base + offset (0x0208)

Description: Error Injection Control 6 (Compare Value Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW3[7:0], TLP_DW3[15:8], TLP_DW3[23:16], TLP_DW3[31:24]
The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the EINJ6_TLP_REG register.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EINJ6_COMPARE_VALUE_H3 Packet Compare Value: 4th DWORD. Specifies the value to compare against Tx the TLP header DWORD#3 bits specified in the Packet Compare Point registers (EINJ6_COMPARE_POINT*). Note: This register field is sticky. |

RAS DES EINJ6 CHANGE POINT H0 REG

Address: Operational Base + offset (0x020c)

Description: Error Injection Control 6 (Change Point Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. It is necessary to carefully consider the

Endianness when you program this register.

Bits [31:0] = TLP_DW0[7:0], TLP_DW0[15:8], TLP_DW0[23:16], TLP_DW0[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_POINT_H0 Packet Change Point: 1st DWORD. Specifies which Tx TLP header DWORD#0 bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). Note: This register field is sticky. |

RAS DES EINJ6 CHANGE POINT H1 REG

Address: Operational Base + offset (0x0210)

Description: Error Injection Control 6 (Change Point Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW1[7:0], TLP_DW1[15:8], TLP_DW1[23:16], TLP_DW1[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_POINT_H1 Packet Change Point: 2nd DWORD. Specifies which Tx TLP header DWORD#1 bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). Note: This register field is sticky. |

RAS DES EINJ6 CHANGE POINT H2 REG

Address: Operational Base + offset (0x0214)

Description: Error Injection Control 6 (Change Point Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW2[7:0], TLP_DW2[15:8], TLP_DW2[23:16], TLP_DW2[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_POINT_H2 Packet Change Point: 3rd DWORD. Specifies which Tx TLP header DWORD#2 bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). Note: This register field is sticky. |

RAS DES EINJ6 CHANGE POINT H3 REG

Address: Operational Base + offset (0x0218)

Description: Error Injection Control 6 (Change Point Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW3[7:0], TLP_DW3[15:8], TLP_DW3[23:16], TLP_DW3[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_POINT_H3 Packet Change Point: 4th DWORD. Specifies which Tx TLP header DWORD#3 bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). Note: This register field is sticky. |

RAS DES EINJ6 CHANGE VALUE H0 REG

Address: Operational Base + offset (0x021c)

Description: Error Injection Control 6 (Change Value Header DWORD #0). Program this register for the 1st DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW0[7:0], TLP_DW0[15:8], TLP_DW0[23:16], TLP_DW0[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_VALUE_H0 Packet Change Value: 1st DWORD. Specifies replacement values for the Tx TLP header DWORD#0 bits defined in the Packet Change Point registers (EINJ6_CHANGE_POINT*). Only applies when the EINJ6_INVERTED_CONTROL field in EINJ6_TLP_REG is '0'. Note: This register field is sticky. |

RAS DES EINJ6 CHANGE VALUE H1 REG

Address: Operational Base + offset (0x0220)

Description: Error Injection Control 6 (Change Value Header DWORD #1). Program this register for the 2nd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW1[7:0], TLP_DW1[15:8], TLP_DW1[23:16], TLP_DW1[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_VALUE_H1 Packet Change Value: 2nd DWORD. Specifies replacement values for the Tx TLP header DWORD#1 bits defined in the Packet Change Point registers (EINJ6_CHANGE_POINT*). Only applies when the EINJ6_INVERTED_CONTROL field in EINJ6_TLP_REG is '0'. Note: This register field is sticky. |

RAS DES EINJ6 CHANGE VALUE H2 REG

Address: Operational Base + offset (0x0224)

Description: Error Injection Control 6 (Change Value Header DWORD #2). Program this register for the 3rd DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW2[7:0], TLP_DW2[15:8], TLP_DW2[23:16], TLP_DW2[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_VALUE_H2 Packet Change Value: 3rd DWORD. Specifies replacement values for the Tx TLP header DWORD#2 bits defined in the Packet Change Point registers (EINJ6_CHANGE_POINT*). Only applies when the EINJ6_INVERTED_CONTROL field in EINJ6_TLP_REG is '0'. Note: This register field is sticky. |

RAS DES EINJ6 CHANGE VALUE H3 REG

Address: Operational Base + offset (0x0228)

Description: Error Injection Control 6 (Change Value Header DWORD #3). Program this register for the 4th DWORD of TLP header/prefix. It is necessary to carefully consider the Endianness when you program this register.

Bits [31:0] = TLP_DW3[7:0], TLP_DW3[15:8], TLP_DW3[23:16], TLP_DW3[31:24]

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the EINJ6_TLP_REG register. Only applies when EINJ6_INVERTED_CONTROL in EINJ6_TLP_REG =0.

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | EINJ6_CHANGE_VALUE_H3 Packet Change Value: 4th DWORD. Specifies replacement values for the Tx TLP header DWORD#3 bits defined in the Packet Change Point registers (EINJ6_CHANGE_POINT*). Only applies when the EINJ6_INVERTED_CONTROL field in EINJ6_TLP_REG is '0'. Note: This register field is sticky. |

RAS DES EINJ6 TLP REG

Address: Operational Base + offset (0x022c)

Description: Error Injection Control 6 (Packet Error).

The Packet Compare Point registers (EINJ6_COMPARE_POINT*) specify which Tx TLP header bits to compare with the corresponding bits in the Packet Compare Value registers (EINJ6_COMPARE_VALUE*). When all specified bits (in the Tx TLP header and EINJ6_COMPARE_VALUE*) match, the controller inserts errors into the TLP. The type and number of errors are specified by the this register.

The Packet Change Point registers (EINJ6_CHANGE_POINT*) specify which Tx TLP header bits to replace with the corresponding bits in the Packet Change Value registers (EINJ6_CHANGE_VALUE*). The type and number of errors are specified by the this register. Only applies when EINJ6_INVERTED_CONTROL in this register =0. The TLP into that errors are injected will not arrive at the transaction layer of the remote device when all of the following conditions are true.

- o Using 128b/130b encoding
- o Injecting errors into TLP Length field / TLP digest bit

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | reserved |
| 11:9 | RW | 0x0 | EINJ6_PACKET_TYPE Packet type. Selects the TLP packets to inject errors into. 0: TLP Header 1: TLP Prefix 1st 4-DWORDs 2: TLP Prefix 2nd -DWORDs Else: Reserved Note: This register field is sticky. |
| 8 | RW | 0x0 | EINJ6_INVERTED_CONTROL Inverted Error Injection Control. 0: EINJ6_CHANGE_VALUE_H[0/1/2/3] is used to replace bits specified by EINJ6_CHANGE_POINT_H[0/1/2/3]. 1: EINJ6_CHANGE_VALUE_H[0/1/2/3] is ignored and inverts bits specified by EINJ6_CHANGE_POINT_H[0/1/2/3]. Note: This register field is sticky. |
| 7:0 | RW | 0x00 | EINJ6_COUNT Error Injection Count. Indicates the number of errors to insert. This counter is decremented while errors are being inserted. If the counter value is 0x01 and error is inserted, ERROR_INJECTION6_ENABLE in EINJ_ENABLE_REG returns '0'. If the counter value is 0x00 and ERROR_INJECTION6_ENABLE=1, errors are inserted until ERROR_INJECTION6_ENABLE is set to '0'. Note: This register field is sticky. |

RAS DES SD CONTROL1 REG

Address: Operational Base + offset (0x0240)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23:22 | RW | 0x0 | LOW_POWER_INTERVAL Low Power Entry Interval Time. Interval Time that the controller starts monitoring RXELECIDLE signal after L0s/L1/L2 entry. You should set the value according to the latency from receiving EIOS to, RXELECIDLE assertion at the PHY. 0x0: 40ns 0x1: 160ns 0x2: 320ns 0x3: 640ns Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 21:20 | RW | 0x0 | <p>TX_EIOS_NUM Number of Tx EIOS.</p> <p>This register sets the number of transmit EIOS for L0s/L1 entry and Disable/Loopback/Hot-reset exit. The controller selects the greater value between this register and the value defined by the PCI-SIG specification.</p> <p>2.5GT/s, 8.0GT/s or higher:</p> <ul style="list-style-type: none"> 0x0: 1 0x1: 4 0x2: 8 0x3: 16 <p>5.0GT/s:</p> <ul style="list-style-type: none"> 0x0: 2 0x1: 8 0x2: 16 0x3: 32 <p>Note: This register field is sticky.</p> |
| 19:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | <p>FORCE_DETECT_LANE_EN Force Detect Lane Enable.</p> <p>When this bit is set, the controller ignores receiver detection from PHY during LTSSM Detect state and uses FORCE_DETECT_LANE.</p> <p>Note: This register field is sticky.</p> |
| 15:0 | RW | 0x0000 | <p>FORCE_DETECT_LANE Force Detect Lane.</p> <p>When the FORCE_DETECT_LANE_EN field is set, the controller ignores receiver detection from PHY during LTSSM Detect state and uses this value instead.</p> <ul style="list-style-type: none"> 0: Lane0 1: Lane1 2: Lane2 .. 15: Lane15 <p>Note: This register field is sticky.</p> |

RAS DES SD CONTROL2 REG

Address: Operational Base + offset (0x0244)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | FRAMING_ERR_RECOVERY_DISABLE Framing Error Recovery Disable. This bit disables a transition to Recovery state when a Framing Error is occurred. Note: This register field is sticky. |
| 15:11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | DIRECT_LPBKSLV_TO_EXIT Direct Loopback Slave To Exit. When this bit is set and the LTSSM is in Loopback Slave Active State, the LTSSM transitions to Loopback Slave Exit state. Note: This register field is sticky. |
| 9 | RW | 0x0 | DIRECT_POLCOMP_TO_DETECT Direct Polling.Compliance to Detect. When this bit is set and the LTSSM is in Polling Compliance State, the LTSSM transitions to Detect state. Note: This register field is sticky. |
| 8 | RW | 0x0 | DIRECT_RECIDLE_TO_CONFIG Direct Recovery.Idle to Configuration. When this bit is set and the LTSSM is in Recovery Idle State, the LTSSM transitions to Configuration state. Note: This register field is sticky. |
| 7:3 | RO | 0x0 | reserved |
| 2 | RW | 0x0 | NOACK_FORCE_LINKDOWN Force LinkDown. When this bit is set and the controller detects REPLY_NUM rolling over 4 times, the LTSSM transitions to Detect State. Note: This register field is sticky. |
| 1 | RW | 0x0 | RECOVERY_REQUEST Recovery Request. When this bit is set to '1' in L0 or L0s, the LTSSM starts transitioning to Recovery State. This request does not cause a speed change or re-equalization. |
| 0 | RW | 0x0 | HOLD_LTSSM Hold and Release LTSSM. |

RAS DES SD STATUS L1LANE REG

Address: Operational Base + offset (0x0250)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x00 | DESKEW_POINTER Deskew Pointer. Indicates Deskew pointer of internal Deskew buffer of selected lane number(LANE_SELECT). Note: This register field is sticky. |
| 23:21 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 20 | RO | 0x1 | PIPE_TXELECIDLE PIPE:TxElecIdle. Indicates PIPE TXELECIDLE signal of selected lane number(LANE_SELECT). Note: This register field is sticky. |
| 19 | RO | 0x1 | PIPE_RXELECIDLE PIPE:RxElecIdle. Indicates PIPE RXELECIDLE signal of selected lane number(LANE_SELECT). Note: This register field is sticky. |
| 18 | RO | 0x0 | PIPE_RXVALID PIPE:RxValid. Indicates PIPE RXVALID signal of selected lane number(LANE_SELECT). Note: This register field is sticky. |
| 17 | RO | 0x0 | PIPE_DETECT_LANE PIPE:Detect Lane. Indicates whether PHY indicates receiver detection or not on selected lane number(LANE_SELECT). Note: This register field is sticky. |
| 16 | RO | 0x0 | PIPE_RXPOLARITY PIPE:RxPolarity. Indicates PIPE RXPOLARITY signal of selected lane number(LANE_SELECT). Note: This register field is sticky. |
| 15:4 | RO | 0x0 | reserved |
| 3:0 | RW | 0x0 | LANE_SELECT Lane Select. Lane Select register for Silicon Debug Status Register of Layer1-PerLane. 0x0: Lane0 0x1: Lane1 0x2: Lane2 .. 0xF: Lane15 Note: This register field is sticky. |

RAS DES SD STATUS L1LTSSM REG

Address: Operational Base + offset (0x0254)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0000 | <p>LTSSM_VARIABLE LTSSM Variable. Indicates internal LTSSM variables defined in the PCI Express Base Specification.</p> <p>C-PCIe Mode:</p> <ul style="list-style-type: none"> 0: directed_speed_change 1: changed_speed_recovery 2: successful_speed_negotiation 3: upconfigure_capable; Set to '1' if both ports advertised the UpConfigure capability in the last Config.Complete. 4: select_deemphasis 5: start_equalization_w_preset 6: equalization_done_8GT_data_rate 7: equalization_done_16GT_data_rate 15:8: idle_to_rlock_transitioned <p>Note: This register field is sticky.</p> |
| 15 | RO | 0x0 | <p>LANE_REVERSAL Lane Reversal Operation. Receiver detected lane reversal. This field is only valid in the L0 LTSSM state.</p> <p>Note: This register field is sticky.</p> |
| 14:11 | RO | 0x0 | reserved |
| 10:8 | RO | 0x2 | <p>PIPE_POWER_DOWN PIPE:PowerDown. Indicates PIPE PowerDown signal.</p> <p>Note: This register field is sticky.</p> |
| 7 | RW | 0x0 | <p>FRAMING_ERR Framing Error. Indicates Framing Error detection status.</p> |

| | | | |
|-----|----|------|--|
| | | | FRAMING_ERR_PTR First Framing Error Pointer. Identifies the first Framing Error using the following encoding. The field contents are only valid value when FRAMING_ERR =1. Received Unexpected Framing Token 01h: When non- STP/SDP/IDL Token was received and it was not in TLP/DLLP reception 02h: When current token was not a valid EDB token and previous token was an EDB. (128/256 bit controller only) 03h: When SDP token was received but not expected. (128 bit & (x8 x16) controller only) 04h: When STP token was received but not expected. (128 bit & (x8 x16) controller only) 05h: When EDS token was expected but not received or whenever an EDS token was received but not expected. 06h: When a framing error was detected in the deskew block while a packet has been in progress in token_finder. Received Unexpected STP Token 11h: When Framing CRC in STP token did not match 12h: When Framing Parity in STP token did not match. 13h: When Framing TLP Length in STP token was smaller than 5 DWORDs. Received Unexpected Block 21h: When Receiving an OS Block following SDS in Datastream state 22h: When Data Block followed by OS Block different from SKP, EI, EIE in Datastream state 23h: When Block with an undefined Block Type in Datastream state 24h: When Data Stream without data over three cycles in Datastream state 25h: When OS Block during Data Stream in Datastream state 26h: When RxStatus Error was detected in Datastream state 27h: When Not all active lanes receiving SKP OS starting at same cycle time in SKPOS state 28h: When a 2-Block timeout occurs for SKP OS in SKPOS state 29h: When Receiving consecutive OS Blocks within a Data Stream in SKPOS state 2Ah: When Phy status error was detected in SKPOS state 2Bh: When Not all active lanes receiving EIOS starting at same cycle time in EIOS state 2Ch: When At least one Symbol from the first 4 Symbols is not EIOS Symbol in EIOS state (CX_NB=2 only) 2Dh: When Not all active lanes receiving EIEOS starting at same cycle time in EIEOS state 2Eh: When Not full 16 eieos symbols are received in EIEOS state |
| 6:0 | RO | 0x00 | |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| | | | All other values not listed above are Reserved. Note: This register field is sticky. |

RAS DES SD STATUS PM REG

Address: Operational Base + offset (0x0258)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RO | 0x00 | LATCHED_NFTS Latched N_FTS. Indicates the value of N_FTS in the received TS Ordered Sets from the Link Partner Note: This register field is sticky. |
| 15:13 | RO | 0x0 | L1SUB_STATE L1Sub State. Indicates internal state machine of L1Sub state. 0h: S_L1_U : idle state 1h: S_L1_0 : wait for aux_clk_active 2h: S_L1_0_WAIT4_ACK : wait for pclkack 3h: S_L1_0_WAIT4_CLKREQ : wait for clkreq 4h: S_L1_N_ENTRY : check clkreq_in_n is de-asserted for t_power_off time (only for L1.2, reduces to one cycle for L1.1) 5h: S_L1_N : L1 substate, turn off txcommonmode circuits (L1.2 only) and rx electrical idle detection circuits 6h: S_L1_N_EXIT : locally/remotely initiated exit, assert pclkreq, wait for pclkack 7h: S_L1_N_ABORT : wait for pclkack when aborting an attempt to enter L1_N Note: This register field is sticky. |
| 12 | RW | 0x0 | PME_RESEND_FLAG PME Re-send flag. When the DUT sends a PM_PME message TLP, the DUT sets PME_Status bit. If host software does not clear PME_Status bit for 100ms(+50%/-5%), the DUT resends the PM_PME Message. This bit indicates that a PM_PME was resent. |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 11:8 | RO | 0x0 | <p>INTERNAL_PM_SSTATE Internal PM State(Slave). Indicates internal state machine of Power Management Slave controller.</p> <p>0h: S_IDLE 1h: S RESPOND_NAK 2h: S_BLOCK_TLP 3h: S_WAIT_LAST_TLP_ACK 4h: S_WAIT_EIDLE 5h: S_LINK_ENTR_L1 6h: S_L1 7h: S_L1_EXIT 8h: S_L23RDY 9h: S_LINK_ENTR_L23 Ah: S_L23RDY_WAIT4ALIVE Bh: S_ACK_WAIT4IDLE Ch: S_WAIT_LAST_PMDLLP Note: This register field is sticky.</p> |
| 7:5 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 4:0 | RO | 0x00 | <p>INTERNAL_PM_MSTATE Internal PM State(Master). Indicates internal state machine of Power Management Master controller.</p> <p>00h: IDLE 01h: L0 02h: L0S 03h: ENTER_L0S 04h: L0S_EXIT 05h: WAIT_PMCSCR_CPL_SENT 08h: L1 09h: L1_BLOCK_TLP 0Ah: L1_WAIT_LAST_TLP_ACK 0Bh: L1_WAIT_PMDLLP_ACK 0Ch: L1_LINK_ENTR_L1 0Dh: L1_EXIT 0Fh: PREP_4L1 10h: L23_BLOCK_TLP 11h: L23_WAIT_LAST_TLP_ACK 12h: L23_WAIT_PMDLLP_ACK 13h: L23_ENTR_L23 14h: L23RDY 15h: PREP_4L23 16h: L23RDY_WAIT4ALIVE 17h: L0S_BLOCK_TLP 18h: WAIT_LAST_PMDLLP 19h: WAIT_DSTATE_UPDATE</p> <p>Note: This register field is sticky.</p> |

RAS DES SD STATUS L2 REG

Address: Operational Base + offset (0x025c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27 | RO | 0x0 | <p>FC_INIT2 FC_INIT2. Indicates the controller is in FC_INIT2(VC0) state. Note: This register field is sticky.</p> |
| 26 | RO | 0x0 | <p>FC_INIT1 FC_INIT1. Indicates the controller is in FC_INIT1(VC0) state. Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 25:24 | RO | 0x0 | DLCMSM DLCMSM. Indicates the current DLCMSM. 00b: DL_INACTIVE 01b: DL_FC_INIT 11b: DL_ACTIVE Note: This register field is sticky. |
| 23:12 | RO | 0xffff | RX_ACK_SEQ_NO Tx Ack Sequence Number. Indicates ACKD_SEQ which is updated by receiving ACK/NAK DLLP. Note: This register field is sticky. |
| 11:0 | RO | 0x000 | TX_TLP_SEQ_NO Tx Tlp Sequence Number. Indicates next transmit sequence number for transmit TLP. Note: This register field is sticky. |

RAS DES SD STATUS L3FC REG

Address: Operational Base + offset (0x0260)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x000 | CREDIT_DATA1 Credit Data1. Current FC credit data selected by the CREDIT_SEL_VC, CREDIT_SEL_CREDIT_TYPE, CREDIT_SEL_TLP_TYPE, and CREDIT_SEL_HD viewport-select fields. Rx: Credit Allocated Value Tx: Credit Limit Value. This value is valid when DLCMSM=0x3(DL_ACTIVE). Note: This register field is sticky. |
| 19:8 | RO | 0x000 | CREDIT_DATA0 Credit Data0. Current FC credit data selected by the CREDIT_SEL_VC, CREDIT_SEL_CREDIT_TYPE, CREDIT_SEL_TLP_TYPE, and CREDIT_SEL_HD viewport-select fields. Rx: Credit Received Value Tx: Credit Consumed Value Note: This register field is sticky. |
| 7 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 6 | RW | 0x0 | CREDIT_SEL_HD Credit Select(HeaderData). This field in conjunction with the CREDIT_SEL_VC, CREDIT_SEL_CREDIT_TYPE, and CREDIT_SEL_TLP_TYPE viewport-select fields determines that data that is returned by the CREDIT_DATA0 and CREDIT_DATA1 data fields. 0x0: Header Credit 0x1: Data Credit Note: This register field is sticky. |
| 5:4 | RW | 0x0 | CREDIT_SEL_TLP_TYPE Credit Select(TLP Type). This field in conjunction with the CREDIT_SEL_VC, CREDIT_SEL_CREDIT_TYPE, and CREDIT_SEL_HD viewport- select fields determines that data that is returned by the CREDIT_DATA0 and CREDIT_DATA1 data fields. 0x0: Posted 0x1: Non-Posted 0x2: Completion Note: This register field is sticky. |
| 3 | RW | 0x0 | CREDIT_SEL_CREDIT_TYPE Credit Select(Credit Type). This field in conjunction with the CREDIT_SEL_VC, CREDIT_SEL_TLP_TYPE, and CREDIT_SEL_HD viewport-select fields determines that data that is returned by the CREDIT_DATA0 and CREDIT_DATA1 data fields. 0x0: Rx 0x1: Tx Note: This register field is sticky. |
| 2:0 | RW | 0x0 | CREDIT_SEL_VC Credit Select(VC). This field in conjunction with the CREDIT_SEL_CREDIT_TYPE, CREDIT_SEL_TLP_TYPE, and CREDIT_SEL_HD viewport-select fields determines that data that is returned by the CREDIT_DATA0 and CREDIT_DATA1 data fields. 0x0: VC0 0x1: VC1 0x2: VC2 .. 0x7: VC7 Note: This register field is sticky. |

RAS DES SD STATUS L3 REG

Address: Operational Base + offset (0x0264)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7 | RW | 0x0 | MFTLP_STATUS Malformed TLP Status. Indicates malformed TLP has occurred. |
| 6:0 | RO | 0x00 | MFTLP_POINTER First Malformed TLP Error Pointer. Indicates the element of the received first malformed TLP. This pointer is validated by MFTLP_STATUS. 01h: AtomicOp address alignment 02h: AtomicOp operand 03h: AtomicOp byte enable 04h: TLP length miss match 05h: Max payload size 06h: Message TLP without TC0 07h: Invalid TC 08h: Unexpected route bit in Message TLP 09h: Unexpected CRS status in Completion TLP 0Ah: Byte enable 0Bh: Memory Address 4KB boundary 0Ch: TLP prefix rules 0Dh: Translation request rules 0Eh: Invalid TLP type 0Fh: Completion rules 7Fh: Application Else: Reserved Note: This register field is sticky. |

RAS DES SD EQ CONTROL1 REG

Address: Operational Base + offset (0x0270)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0x00 | FOM_TARGET FOM Target. Indicates figure of merit target criteria value of EQ Master(DSP in EQ Phase3/USP in EQ Phase2). This field is only valid when GEN3_EQ_FB_MODE is 0001b(Figure Of Merit). Note: This register field is sticky. |
| 23 | RW | 0x0 | FOM_TARGET_ENABLE FOM Target Enable. Enables the FOM_TARGET fields. |
| 22:18 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17:16 | RW | 0x0 | <p>EVAL_INTERVAL_TIME Eval Interval Time. Indicates interval time of RxEqEval assertion.</p> <p>00: 500ns 01: 1us 10: 2us 11: 4us</p> <p>This field is used for EQ Master(DSP in EQ Phase3/USP in EQ Phase2).</p> <p>Note: This register field is sticky.</p> |
| 15:10 | RO | 0x0 | reserved |
| 9:8 | RW | 0x0 | <p>EXT_EQ_TIMEOUT Extends EQ Phase2/3 Timeout.</p> <p>This field is used when the Ltssm is in Recovery.EQ2/3. When this field is set, the value of EQ2/3 timeout is extended. EQ Master(DSP in EQ Phase3/USP in EQ Phase2).</p> <p>00: 24ms (default) 01: 48ms (x2) 10: 240ms (x10) 11: No timeout</p> <p>EQ Slave(DSP in EQ Phase2/USP in EQ Phase3).</p> <p>00: 32ms (default) 01: 56ms (32ms+24ms) 10: 248ms (32ms +9*24ms) 11: No timeout</p> <p>Note: This register field is sticky.</p> |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | <p>EQ_RATE_SEL EQ Status Rate Select.</p> <p>Setting this field in conjunction with the EQ_LANE_SEL field determines the per-lane Silicon Debug EQ Status data returned by the SD_EQ_CONTROL[2/3] and SD_EQ_STATUS[1/2/3] viewport registers.</p> <p>0x0: 8.0GT/s Speed 0x1: 16.0GT/s Speed</p> <p>Note: This register field is sticky.</p> |
| 3:0 | RW | 0x0 | <p>EQ_LANE_SEL EQ Status Lane Select.</p> <p>Setting this field in conjunction with the EQ_RATE_SEL field determines the per-lane Silicon Debug EQ Status data returned by the SD_EQ_CONTROL[2/3] and SD_EQ_STATUS[1/2/3] viewport registers.</p> <p>0x0: Lane0 0x1: Lane1</p> <p>Note: This register field is sticky.</p> |

RAS DES SD EQ CONTROL2 REG

Address: Operational Base + offset (0x0274)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30 | RW | 0x0 | FORCE_LOCAL_TX_PRESET_ENABLE Force Local Transmitter Preset Enable. Enables the FORCE_LOCAL_TX_PRESET field. Note: This register field is sticky. |
| 29 | RW | 0x0 | FORCE_LOCAL_RX_HINT_ENABLE Force Local Receiver Preset Hint Enable. Enables the FORCE_LOCAL_RX_HINT field. Note: This register field is sticky. |
| 28 | RW | 0x0 | FORCE_LOCAL_TX_COEF_ENABLE Force Local Transmitter Coefficient Enable. Enables the following fields: FORCE_LOCAL_TX_PRE_CURSOR FORCE_LOCAL_TX_CURSOR FORCE_LOCAL_TX_POST_CURSOR Note: This register field is sticky. |
| 27:24 | RW | 0x0 | FORCE_LOCAL_TX_PRESET Force Local Transmitter Preset. Indicates initial preset value of USP in EQ Slave(EQ Phase2) instead of receiving EQ TS2. Note: This register field is sticky. |
| 23:21 | RO | 0x0 | reserved |
| 20:18 | RW | 0x0 | FORCE_LOCAL_RX_HINT Force Local Receiver Preset Hint. Indicates the RxPresetHint value of EQ Slave(DSP in EQ Phase2/USP in EQ Phase3), instead of received or set value. Note: This register field is sticky. |
| 17:12 | RW | 0x00 | FORCE_LOCAL_TX_POST_CURSOR Force Local Transmitter Post-Cursor. Indicates the coefficient value of EQ Slave(DSP in EQ Phase2/USP in EQ Phase3), instead of the value instructed from link partner. Note: This register field is sticky. |
| 11:6 | RW | 0x00 | FORCE_LOCAL_TX_CURSOR Force Local Transmitter Cursor. Indicates the coefficient value of EQ Slave(DSP in EQ Phase2/USP in EQ Phase3), instead of the value instructed from link partner. Note: This register field is sticky. |
| 5:0 | RW | 0x00 | FORCE_LOCAL_TX_PRE_CURSOR Force Local Transmitter Pre-cursor. Indicates the coefficient value of EQ Slave(DSP in EQ Phase2/USP in EQ Phase3), instead of the value instructed from link partner. Note: This register field is sticky. |

RAS DES SD EQ CONTROL3 REG

Address: Operational Base + offset (0x0278)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:29 | RO | 0x0 | reserved |
| 28 | RW | 0x0 | <p>FORCE_REMOTE_TX_COEF_ENABLE Force Remote Transmitter Coefficient Enable. Enables the following fields:</p> <p>FORCE_REMOTE_TX_PRE_CURSOR FORCE_REMOTE_TX_CURSOR FORCE_REMOTE_TX_POST_CURSOR</p> <p>Note: This register field is sticky.</p> |
| 27:18 | RO | 0x0 | reserved |
| 17:12 | RW | 0x00 | <p>FORCE_REMOTE_TX_POST_CURSOR Force Remote Transmitter Post-Cursor. Indicates the coefficient value of EQ Master(DSP in EQ Phase3/USP in EQ Phase2), instead of the value instructed from link partner.</p> <p>Note: This register field is sticky.</p> |
| 11:6 | RW | 0x00 | <p>FORCE_REMOTE_TX_CURSOR Force Remote Transmitter Cursor. Indicates the coefficient value of EQ Master(DSP in EQ Phase3/USP in EQ Phase2), instead of the value instructed from link partner.</p> <p>Note: This register field is sticky.</p> |
| 5:0 | RW | 0x00 | <p>FORCE_REMOTE_TX_PRE_CURSOR Force Remote Transmitter Pre-Cursor. Indicates the coefficient value of EQ Master(DSP in EQ Phase3/USP in EQ Phase2), instead of the value instructed from link partner.</p> <p>Note: This register field is sticky.</p> |

RAS DES SD EQ STATUS1 REG

Address: Operational Base + offset (0x0280)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7 | RO | 0x0 | <p>EQ_REJECT_EVENT EQ Reject Event. Indicates that the controller receives two consecutive TS1 OS w/Reject=1b during EQ Master phase(DSP in EQ Phase3/USP in EQ Phase2). This bit is automatically cleared when the controller starts EQ Master phase again.</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 6 | RO | 0x0 | <p>EQ_RULEC_VIOLATION EQ Rule C Violation.</p> <p>Indicates that coefficients rule C violation is detected in the values provided by PHY using direction change method during EQ Master phase(DSP in EQ Phase3/USP in EQ Phase2). The coefficients rule C correspond to the rules c) from section "Rules for Transmitter Coefficents" in the PCI Express Base Specification.</p> <p>This bit is automatically cleared when the controller starts EQ Master phase again.</p> <p>Note: This register field is sticky.</p> |
| 5 | RO | 0x0 | <p>EQ_RULEB_VIOLATION EQ Rule B Violation.</p> <p>Indicates that coefficients rule B violation is detected in the values provided by PHY using direction change method during EQ Master phase(DSP in EQ Phase3/USP in EQ Phase2). The coefficients rules B correspond to the rules b) from section "Rules for Transmitter Coefficents" in the PCI Express Base Specification.</p> <p>This bit is automatically cleared when the controller starts EQ Master phase again.</p> <p>Note: This register field is sticky.</p> |
| 4 | RO | 0x0 | <p>EQ_RULEA_VIOLATION EQ Rule A Violation.</p> <p>Indicates that coefficients rule A violation is detected in the values provided by PHY using direction change method during EQ Master phase(DSP in EQ Phase3/USP in EQ Phase2). The coefficients rules A correspond to the rules a) from section "Rules for Transmitter Coefficents" in the PCI Express Base Specification.</p> <p>This bit is automatically cleared when the controller starts EQ Master phase again.</p> <p>Note: This register field is sticky.</p> |
| 3 | RO | 0x0 | reserved |
| 2:1 | RO | 0x0 | <p>EQ_CONVERGENCE_INFO EQ Convergence Info.</p> <p>Indicates equalization convergence information.</p> <p>0x0: Equalization is not attempted 0x1: Equalization finished successfully 0x2: Equalization finished unsuccessfully 0x3: Reserved</p> <p>This bit is automatically cleared when the controller starts EQ Master phase again.</p> <p>Note: This register field is sticky.</p> |
| 0 | RO | 0x0 | <p>EQ_SEQUENCE EQ Sequence.</p> <p>Indicates that the controller is starting the equalization sequence.</p> <p>Note: This register field is sticky.</p> |

RAS DES SD EQ STATUS2 REG

Address: Operational Base + offset (0x0284)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x00 | EQ_LOCAL_FOM_VALUE EQ Local Figure of Merit. Indicates Local maximum Figure of Merit value. Note: This register field is sticky. |
| 23:21 | RO | 0x0 | reserved |
| 20:18 | RO | 0x0 | EQ_LOCAL_RX_HINT EQ Local Receiver Preset Hint. Indicates Local Receiver Preset Hint value. Note: This register field is sticky. |
| 17:12 | RO | 0x00 | EQ_LOCAL_POST_CURSOR EQ Local Post-Cursor. Indicates Local post cursor coefficient value. Note: This register field is sticky. |
| 11:6 | RO | 0x00 | EQ_LOCAL_CURSOR EQ Local Cursor. Indicates Local cursor coefficient value. Note: This register field is sticky. |
| 5:0 | RO | 0x00 | EQ_LOCAL_PRE_CURSOR EQ Local Pre-Cursor. Indicates Local pre cursor coefficient value. Note: This register field is sticky. |

RAS DES SD EQ STATUS3 REG

Address: Operational Base + offset (0x0288)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | reserved |
| 29:24 | RO | 0x00 | EQ_REMOTE_FS EQ Remote FS. Indicates Remote FS value. Note: This register field is sticky. |
| 23:18 | RO | 0x00 | EQ_REMOTE_LF EQ Remote LF. Indicates Remote LF value. Note: This register field is sticky. |
| 17:12 | RO | 0x00 | EQ_REMOTE_POST_CURSOR EQ Remote Post-Cursor. Indicates Remote post cursor coefficient value. Note: This register field is sticky. |
| 11:6 | RO | 0x00 | EQ_REMOTE_CURSOR EQ Remote Cursor. Indicates Remote cursor coefficient value. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5:0 | RO | 0x00 | EQ_REMOTE_PRE_CURSOR EQ Remote Pre-Cursor. Indicates Remote pre cursor coefficient value. Note: This register field is sticky. |

10.5.28 Resizable Bar Capability Registers Summary

Resizable BAR Capability Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|---------------------------------------|--------|------|-------------|-------------------------------------|
| RESBAR_CAP_HDR_REG | 0x02a0 | W | 0x00010015 | Resizable BAR Capability Header. |
| RESBAR_CAP_REG_0_REG | 0x02a4 | W | 0x00000010 | Resizable BAR0 Capability Register. |
| RESBAR_CTRL_REG_0_REG | 0x02a8 | W | 0x000000c0 | Resizable BAR0 Control Register. |
| RESBAR_CAP_REG_1_REG | 0x02ac | W | 0x00000010 | Resizable BAR1 Capability Register. |
| RESBAR_CTRL_REG_1_REG | 0x02b0 | W | 0x00000000 | Resizable BAR1 Control Register. |
| RESBAR_CAP_REG_2_REG | 0x02b4 | W | 0x00000010 | Resizable BAR2 Capability Register. |
| RESBAR_CTRL_REG_2_REG | 0x02b8 | W | 0x00000000 | Resizable BAR2 Control Register. |
| RESBAR_CAP_REG_3_REG | 0x02bc | W | 0x00000010 | Resizable BAR3 Capability Register. |
| RESBAR_CTRL_REG_3_REG | 0x02c0 | W | 0x00000000 | Resizable BAR3 Control Register. |
| RESBAR_CAP_REG_4_REG | 0x02c4 | W | 0x00000010 | Resizable BAR4 Capability Register. |
| RESBAR_CTRL_REG_4_REG | 0x02c8 | W | 0x00000000 | Resizable BAR4 Control Register. |
| RESBAR_CAP_REG_5_REG | 0x02cc | W | 0x00000010 | Resizable BAR5 Capability Register. |
| RESBAR_CTRL_REG_5_REG | 0x02d0 | W | 0x00000000 | Resizable BAR5 Control Register. |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.29 Resizable Bar Capability Detail Register Description

[RESBAR_CAP_HDR_REG](#)

Address: Operational Base + offset (0x02a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x000 | <p>RESBAR_CAP_NEXT_OFFSET Next Capability Offset. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19:16 | RO | 0x1 | <p>RESBAR_CAP_VERSION Capability Version. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15:0 | RO | 0x0015 | <p>RESBAR_EXT_CAP_ID Resizable BAR Extended Capability ID. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

RESBAR CAP REG_0 REG

Address: Operational Base + offset (0x02a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | <p>RESBAR_CAP_REG_0_128TB Up to 128TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CAP_REG_0_64TB Up to 64TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | <p>RESBAR_CAP_REG_0_32TB Up to 32TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 28 | RW | 0x0 | <p>RESBAR_CAP_REG_0_16TB Up to 16TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CAP_REG_0_8TB Up to 8TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CAP_REG_0_4TB Up to 4TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CAP_REG_0_2TB Up to 2TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CAP_REG_0_1TB Up to 1TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23 | RW | 0x0 | <p>RESBAR_CAP_REG_0_512GB Up to 512GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 22 | RW | 0x0 | <p>RESBAR_CAP_REG_0_256GB Up to 256GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CAP_REG_0_128GB Up to 128B BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CAP_REG_0_64GB Up to 64GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CAP_REG_0_32GB Up to 32GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CAP_REG_0_16GB Up to 16GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17 | RW | 0x0 | <p>RESBAR_CAP_REG_0_8GB Up to 8GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 16 | RW | 0x0 | <p>RESBAR_CAP_REG_0_4GB Up to 4GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15 | RW | 0x0 | <p>RESBAR_CAP_REG_0_2GB Up to 2GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 14 | RW | 0x0 | <p>RESBAR_CAP_REG_0_1GB Up to 1GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 13 | RW | 0x0 | <p>RESBAR_CAP_REG_0_512MB Up to 512MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 12 | RW | 0x0 | <p>RESBAR_CAP_REG_0_256MB Up to 256MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | <p>RESBAR_CAP_REG_0_128MB Up to 128MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 10 | RW | 0x0 | <p>RESBAR_CAP_REG_0_64MB Up to 64MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>RESBAR_CAP_REG_0_32MB Up to 32MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 8 | RW | 0x0 | <p>RESBAR_CAP_REG_0_16MB Up to 16MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 7 | RW | 0x0 | <p>RESBAR_CAP_REG_0_8MB Up to 8MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 6 | RW | 0x0 | <p>RESBAR_CAP_REG_0_4MB Up to 4MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | RW | 0x0 | <p>RESBAR_CAP_REG_0_2MB Up to 2MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 4 | RW | 0x1 | <p>RESBAR_CAP_REG_0_1MB Up to 1MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 3:0 | RO | 0x0 | reserved |

RESBAR_CTRL_REG_0_REG

Address: Operational Base + offset (0x02a8)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 31 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_8EB Up to 8EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_4EB Up to 4EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 29 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_2EB Up to 2EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_1EB Up to 1EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_512PB Up to 512PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_256PB Up to 256PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_128PB Up to 128PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_64PB Up to 64PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 23 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_32PB Up to 32PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 22 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_16PB Up to 16PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_8PB Up to 8PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_4PB Up to 4PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_2PB Up to 2PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_1PB Up to 1PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 17 | RW | 0x0 | <p>RESBAR_CTRL_REG_0_512TB Up to 512TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | RW | 0x0 | RESBAR_CTRL_REG_0_256TB Up to 256TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | RESBAR_CTRL_REG_BAR_SIZE BAR Size. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R/W (sticky) Dbi: R/W (sticky) Note: This register field is sticky. |
| 7:5 | RO | 0x6 | RESBAR_CTRL_REG_NUM_BARS Number of Resizable BARs. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |
| 4:3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | RESBAR_CTRL_REG_IDX_0 BAR Index. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |

RESBAR_CAP_REG_1_REG

Address: Operational Base + offset (0x02ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | RESBAR_CAP_REG_1_128TB Up to 128TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 30 | RW | 0x0 | RESBAR_CAP_REG_1_64TB Up to 64TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | <p>RESBAR_CAP_REG_1_32TB Up to 32TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 28 | RW | 0x0 | <p>RESBAR_CAP_REG_1_16TB Up to 16TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CAP_REG_1_8TB Up to 8TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CAP_REG_1_4TB Up to 4TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CAP_REG_1_2TB Up to 2TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CAP_REG_1_1TB Up to 1TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23 | RW | 0x0 | <p>RESBAR_CAP_REG_1_512GB Up to 512GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 22 | RW | 0x0 | <p>RESBAR_CAP_REG_1_256GB Up to 256GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CAP_REG_1_128GB Up to 128GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CAP_REG_1_64GB Up to 64GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CAP_REG_1_32GB Up to 32GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CAP_REG_1_16GB Up to 16GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17 | RW | 0x0 | <p>RESBAR_CAP_REG_1_8GB Up to 8GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 16 | RW | 0x0 | <p>RESBAR_CAP_REG_1_4GB Up to 4GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15 | RW | 0x0 | <p>RESBAR_CAP_REG_1_2GB Up to 2GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 14 | RW | 0x0 | <p>RESBAR_CAP_REG_1_1GB Up to 1GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 13 | RW | 0x0 | <p>RESBAR_CAP_REG_1_512MB Up to 512MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 12 | RW | 0x0 | <p>RESBAR_CAP_REG_1_256MB Up to 256MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | <p>RESBAR_CAP_REG_1_128MB Up to 128MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 10 | RW | 0x0 | <p>RESBAR_CAP_REG_1_64MB Up to 64MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>RESBAR_CAP_REG_1_32MB Up to 32MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 8 | RW | 0x0 | <p>RESBAR_CAP_REG_1_16MB Up to 16MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 7 | RW | 0x0 | <p>RESBAR_CAP_REG_1_8MB Up to 8MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 6 | RW | 0x0 | <p>RESBAR_CAP_REG_1_4MB Up to 4MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | RW | 0x0 | <p>RESBAR_CAP_REG_1_2MB Up to 2MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 4 | RW | 0x1 | <p>RESBAR_CAP_REG_1_1MB Up to 1MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 3:0 | RO | 0x0 | reserved |

RESBAR_CTRL_REG_1_REG

Address: Operational Base + offset (0x02b0)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 31 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_8EB Up to 8EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_4EB Up to 4EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 29 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_2EB Up to 2EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_1EB Up to 1EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_512PB Up to 512PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_256PB Up to 256PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_128PB Up to 128PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_64PB Up to 64PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 23 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_32PB Up to 32PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 22 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_16PB Up to 16PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_8PB Up to 8PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_4PB Up to 4PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_2PB Up to 2PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_1PB Up to 1PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 17 | RW | 0x0 | <p>RESBAR_CTRL_REG_1_512TB Up to 512TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | RW | 0x0 | RESBAR_CTRL_REG_1_256TB Up to 256TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | RESBAR_CTRL_REG_BAR_SIZE BAR Size. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R/W (sticky) Dbi: R/W (sticky) Note: This register field is sticky. |
| 7:5 | RO | 0x0 | RESBAR_CTRL_REG_NUM_BARS Number of Resizable BARs. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |
| 4:3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | RESBAR_CTRL_REG_IDX_1 BAR Index. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |

RESBAR_CAP_REG_2_REG

Address: Operational Base + offset (0x02b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | RESBAR_CAP_REG_2_128TB Up to 128TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 30 | RW | 0x0 | RESBAR_CAP_REG_2_64TB Up to 64TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | <p>RESBAR_CAP_REG_2_32TB Up to 32TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 28 | RW | 0x0 | <p>RESBAR_CAP_REG_2_16TB Up to 16TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CAP_REG_2_8TB Up to 8TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CAP_REG_2_4TB Up to 4TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CAP_REG_2_2TB Up to 2TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CAP_REG_2_1TB Up to 1TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23 | RW | 0x0 | RESBAR_CAP_REG_2_512GB Up to 512GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 22 | RW | 0x0 | RESBAR_CAP_REG_2_256GB Up to 256GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 21 | RW | 0x0 | RESBAR_CAP_REG_2_128GB Up to 128GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 20 | RW | 0x0 | RESBAR_CAP_REG_2_64GB Up to 64GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 19 | RW | 0x0 | RESBAR_CAP_REG_2_32GB Up to 32GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 18 | RW | 0x0 | RESBAR_CAP_REG_2_16GB Up to 16GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17 | RW | 0x0 | <p>RESBAR_CAP_REG_2_8GB Up to 8GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 16 | RW | 0x0 | <p>RESBAR_CAP_REG_2_4GB Up to 4GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15 | RW | 0x0 | <p>RESBAR_CAP_REG_2_2GB Up to 2GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 14 | RW | 0x0 | <p>RESBAR_CAP_REG_2_1GB Up to 1GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 13 | RW | 0x0 | <p>RESBAR_CAP_REG_2_512MB Up to 512MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 12 | RW | 0x0 | <p>RESBAR_CAP_REG_2_256MB Up to 256MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | <p>RESBAR_CAP_REG_2_128MB Up to 128MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 10 | RW | 0x0 | <p>RESBAR_CAP_REG_2_64MB Up to 64MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>RESBAR_CAP_REG_2_32MB Up to 32MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 8 | RW | 0x0 | <p>RESBAR_CAP_REG_2_16MB Up to 16MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 7 | RW | 0x0 | <p>RESBAR_CAP_REG_2_8MB Up to 8MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 6 | RW | 0x0 | <p>RESBAR_CAP_REG_2_4MB Up to 4MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | RW | 0x0 | <p>RESBAR_CAP_REG_2_2MB Up to 2MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 4 | RW | 0x1 | <p>RESBAR_CAP_REG_2_1MB Up to 1MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 3:0 | RO | 0x0 | reserved |

RESBAR_CTRL_REG_2_REG

Address: Operational Base + offset (0x02b8)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 31 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_8EB Up to 8EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_4EB Up to 4EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 29 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_2EB Up to 2EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_1EB Up to 1EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_512PB Up to 512PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_256PB Up to 256PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_128PB Up to 128PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_64PB Up to 64PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 23 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_32PB Up to 32PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 22 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_16PB Up to 16PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_8PB Up to 8PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_4PB Up to 4PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_2PB Up to 2PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_1PB Up to 1PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 17 | RW | 0x0 | <p>RESBAR_CTRL_REG_2_512TB Up to 512TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | RW | 0x0 | RESBAR_CTRL_REG_2_256TB Up to 256TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | RESBAR_CTRL_REG_BAR_SIZE BAR Size. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R/W (sticky) Dbi: R/W (sticky) Note: This register field is sticky. |
| 7:5 | RO | 0x0 | RESBAR_CTRL_REG_NUM_BARS Number of Resizable BARs. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |
| 4:3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | RESBAR_CTRL_REG_IDX_2 BAR Index. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |

RESBAR_CAP_REG_3_REG

Address: Operational Base + offset (0x02bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | RESBAR_CAP_REG_3_128TB Up to 128TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 30 | RW | 0x0 | RESBAR_CAP_REG_3_64TB Up to 64TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | <p>RESBAR_CAP_REG_3_32TB Up to 32TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 28 | RW | 0x0 | <p>RESBAR_CAP_REG_3_16TB Up to 16TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CAP_REG_3_8TB Up to 8TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CAP_REG_3_4TB Up to 4TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CAP_REG_3_2TB Up to 2TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CAP_REG_3_1TB Up to 1TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23 | RW | 0x0 | RESBAR_CAP_REG_3_512GB Up to 512GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 22 | RW | 0x0 | RESBAR_CAP_REG_3_256GB Up to 256GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 21 | RW | 0x0 | RESBAR_CAP_REG_3_128GB Up to 128GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 20 | RW | 0x0 | RESBAR_CAP_REG_3_64GB Up to 64GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 19 | RW | 0x0 | RESBAR_CAP_REG_3_32GB Up to 32GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 18 | RW | 0x0 | RESBAR_CAP_REG_3_16GB Up to 16GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17 | RW | 0x0 | <p>RESBAR_CAP_REG_3_8GB Up to 8GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 16 | RW | 0x0 | <p>RESBAR_CAP_REG_3_4GB Up to 4GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15 | RW | 0x0 | <p>RESBAR_CAP_REG_3_2GB Up to 2GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 14 | RW | 0x0 | <p>RESBAR_CAP_REG_3_1GB Up to 1GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 13 | RW | 0x0 | <p>RESBAR_CAP_REG_3_512MB Up to 512MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 12 | RW | 0x0 | <p>RESBAR_CAP_REG_3_256MB Up to 256MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | <p>RESBAR_CAP_REG_3_128MB Up to 128MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 10 | RW | 0x0 | <p>RESBAR_CAP_REG_3_64MB Up to 64MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>RESBAR_CAP_REG_3_32MB Up to 32MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 8 | RW | 0x0 | <p>RESBAR_CAP_REG_3_16MB Up to 16MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 7 | RW | 0x0 | <p>RESBAR_CAP_REG_3_8MB Up to 8MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 6 | RW | 0x0 | <p>RESBAR_CAP_REG_3_4MB Up to 4MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | RW | 0x0 | <p>RESBAR_CAP_REG_3_2MB Up to 2MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 4 | RW | 0x1 | <p>RESBAR_CAP_REG_3_1MB Up to 1MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 3:0 | RO | 0x0 | reserved |

RESBAR_CTRL_REG_3_REG

Address: Operational Base + offset (0x02c0)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 31 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_8EB Up to 8EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_4EB Up to 4EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 29 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_2EB Up to 2EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_1EB Up to 1EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_512PB Up to 512PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_256PB Up to 256PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_128PB Up to 128PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_64PB Up to 64PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 23 | RW | 0x0 | <p>RESBAR_CTRL_REG_3_32PB Up to 32PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 22 | RW | 0x0 | RESBAR_CTRL_REG_3_16PB Up to 16PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 21 | RW | 0x0 | RESBAR_CTRL_REG_3_8PB Up to 8PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 20 | RW | 0x0 | RESBAR_CTRL_REG_3_4PB Up to 4PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 19 | RW | 0x0 | RESBAR_CTRL_REG_3_2PB Up to 2PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 18 | RW | 0x0 | RESBAR_CTRL_REG_3_1PB Up to 1PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 17 | RW | 0x0 | RESBAR_CTRL_REG_3_512TB Up to 512TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | RW | 0x0 | RESBAR_CTRL_REG_3_256TB Up to 256TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | RESBAR_CTRL_REG_BAR_SIZE BAR Size. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R/W (sticky) Dbi: R/W (sticky) Note: This register field is sticky. |
| 7:5 | RO | 0x0 | RESBAR_CTRL_REG_NUM_BARS Number of Resizable BARs. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |
| 4:3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | RESBAR_CTRL_REG_IDX_3 BAR Index. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |

RESBAR_CAP_REG_4_REG

Address: Operational Base + offset (0x02c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | RESBAR_CAP_REG_4_128TB Up to 128TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 30 | RW | 0x0 | RESBAR_CAP_REG_4_64TB Up to 64TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | <p>RESBAR_CAP_REG_4_32TB Up to 32TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 28 | RW | 0x0 | <p>RESBAR_CAP_REG_4_16TB Up to 16TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CAP_REG_4_8TB Up to 8TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CAP_REG_4_4TB Up to 4TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CAP_REG_4_2TB Up to 2TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CAP_REG_4_1TB Up to 1TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23 | RW | 0x0 | <p>RESBAR_CAP_REG_4_512GB Up to 512GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 22 | RW | 0x0 | <p>RESBAR_CAP_REG_4_256GB Up to 256GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CAP_REG_4_128GB Up to 128GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CAP_REG_4_64GB Up to 64GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CAP_REG_4_32GB Up to 32GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CAP_REG_4_16GB Up to 16GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17 | RW | 0x0 | <p>RESBAR_CAP_REG_4_8GB Up to 8GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 16 | RW | 0x0 | <p>RESBAR_CAP_REG_4_4GB Up to 4GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15 | RW | 0x0 | <p>RESBAR_CAP_REG_4_2GB Up to 2GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 14 | RW | 0x0 | <p>RESBAR_CAP_REG_4_1GB Up to 1GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 13 | RW | 0x0 | <p>RESBAR_CAP_REG_4_512MB Up to 512MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 12 | RW | 0x0 | <p>RESBAR_CAP_REG_4_256MB Up to 256MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | <p>RESBAR_CAP_REG_4_128MB Up to 128MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 10 | RW | 0x0 | <p>RESBAR_CAP_REG_4_64MB Up to 64MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>RESBAR_CAP_REG_4_32MB Up to 32MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 8 | RW | 0x0 | <p>RESBAR_CAP_REG_4_16MB Up to 16MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 7 | RW | 0x0 | <p>RESBAR_CAP_REG_4_8MB Up to 8MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 6 | RW | 0x0 | <p>RESBAR_CAP_REG_4_4MB Up to 4MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | RW | 0x0 | <p>RESBAR_CAP_REG_4_2MB Up to 2MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 4 | RW | 0x1 | <p>RESBAR_CAP_REG_4_1MB Up to 1MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 3:0 | RO | 0x0 | reserved |

RESBAR_CTRL_REG_4_REG

Address: Operational Base + offset (0x02c8)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 31 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_8EB Up to 8EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_4EB Up to 4EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 29 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_2EB Up to 2EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_1EB Up to 1EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_512PB Up to 512PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_256PB Up to 256PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_128PB Up to 128PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_64PB Up to 64PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 23 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_32PB Up to 32PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 22 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_16PB Up to 16PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_8PB Up to 8PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_4PB Up to 4PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_2PB Up to 2PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_1PB Up to 1PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 17 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_512TB Up to 512TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | RW | 0x0 | <p>RESBAR_CTRL_REG_4_256TB Up to 256TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | <p>RESBAR_CTRL_REG_BAR_SIZE BAR Size. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W (sticky) Dbi: R/W (sticky)</p> <p>Note: This register field is sticky.</p> |
| 7:5 | RO | 0x0 | <p>RESBAR_CTRL_REG_NUM_BARS Number of Resizable BARs. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: This register field is sticky.</p> |
| 4:3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | <p>RESBAR_CTRL_REG_IDX_4 BAR Index. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: This register field is sticky.</p> |

RESBAR_CAP_REG_5_REG

Address: Operational Base + offset (0x02cc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | <p>RESBAR_CAP_REG_5_128TB Up to 128TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CAP_REG_5_64TB Up to 64TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | <p>RESBAR_CAP_REG_5_32TB Up to 32TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 28 | RW | 0x0 | <p>RESBAR_CAP_REG_5_16TB Up to 16TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CAP_REG_5_8TB Up to 8TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CAP_REG_5_4TB Up to 4TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CAP_REG_5_2TB Up to 2TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CAP_REG_5_1TB Up to 1TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23 | RW | 0x0 | <p>RESBAR_CAP_REG_5_512GB Up to 512GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 22 | RW | 0x0 | <p>RESBAR_CAP_REG_5_256GB Up to 256GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CAP_REG_5_128GB Up to 128GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CAP_REG_5_64GB Up to 64GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CAP_REG_5_32GB Up to 32GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CAP_REG_5_16GB Up to 16GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17 | RW | 0x0 | <p>RESBAR_CAP_REG_5_8GB Up to 8GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 16 | RW | 0x0 | <p>RESBAR_CAP_REG_5_4GB Up to 4GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 15 | RW | 0x0 | <p>RESBAR_CAP_REG_5_2GB Up to 2GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 14 | RW | 0x0 | <p>RESBAR_CAP_REG_5_1GB Up to 1GB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 13 | RW | 0x0 | <p>RESBAR_CAP_REG_5_512MB Up to 512MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 12 | RW | 0x0 | <p>RESBAR_CAP_REG_5_256MB Up to 256MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | <p>RESBAR_CAP_REG_5_128MB Up to 128MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 10 | RW | 0x0 | <p>RESBAR_CAP_REG_5_64MB Up to 64MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>RESBAR_CAP_REG_5_32MB Up to 32MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 8 | RW | 0x0 | <p>RESBAR_CAP_REG_5_16MB Up to 16MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 7 | RW | 0x0 | <p>RESBAR_CAP_REG_5_8MB Up to 8MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 6 | RW | 0x0 | <p>RESBAR_CAP_REG_5_4MB Up to 4MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | RW | 0x0 | <p>RESBAR_CAP_REG_5_2MB Up to 2MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 4 | RW | 0x1 | <p>RESBAR_CAP_REG_5_1MB Up to 1MB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 3:0 | RO | 0x0 | reserved |

RESBAR_CTRL_REG_5_REG

Address: Operational Base + offset (0x02d0)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 31 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_8EB Up to 8EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_4EB Up to 4EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 29 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_2EB Up to 2EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 28 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_1EB Up to 1EB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_512PB Up to 512PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_256PB Up to 256PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_128PB Up to 128PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_64PB Up to 64PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 23 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_32PB Up to 32PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 22 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_16PB Up to 16PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 21 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_8PB Up to 8PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 20 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_4PB Up to 4PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 19 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_2PB Up to 2PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 18 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_1PB Up to 1PB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |
| 17 | RW | 0x0 | <p>RESBAR_CTRL_REG_5_512TB Up to 512TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification.</p> <p>Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky)</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 16 | RW | 0x0 | RESBAR_CTRL_REG_5_256TB Up to 256TB BAR Supported. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R (sticky) Dbi: if (DBI_RO_WR_EN == 1) then R/W(sticky) else R(sticky) Note: This register field is sticky. |
| 15:14 | RO | 0x0 | reserved |
| 13:8 | RW | 0x00 | RESBAR_CTRL_REG_BAR_SIZE BAR Size. For a description of this standard PCIe register field, see the PCI Express Specification. Note: The access attributes of this field are as follows: Wire: R/W (sticky) Dbi: R/W (sticky) Note: This register field is sticky. |
| 7:5 | RO | 0x0 | RESBAR_CTRL_REG_NUM_BARS Number of Resizable BARs. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |
| 4:3 | RO | 0x0 | reserved |
| 2:0 | RO | 0x0 | RESBAR_CTRL_REG_IDX_5 BAR Index. For a description of this standard PCIe register field, see the PCI Express Specification. Note: This register field is sticky. |

10.5.30 Port Logic Registers Summary

Port Logic Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|--------------------------|--------|------|-------------|---|
| PL_ACK_LATENCY_TIMER_OFF | 0x0700 | W | 0x18460817 | Ack Latency Timer and Replay Timer Register. |
| PL_VENDOR_SPEC_DLLP_OFF | 0x0704 | W | 0xffffffff | Vendor Specific DLLP Register. |
| PL_PORT_FORCE_OFF | 0x0708 | W | 0x00000004 | PORT_FORCE_OFF |
| PL_ACK_F_ASMP_CTRL_OF | 0x070c | W | 0x1bffff00 | Ack Frequency and L0-L1 ASMP Control Register. |
| PL_PORT_LINK_CTRL_OF | 0x0710 | W | 0x00030120 | Port Link Control Register. |
| PL_LANE_SKEW_OF | 0x0714 | W | 0x08000000 | Lane Skew Register. |
| PL_TIMER_CTRL_MAX_FU_N | 0x0718 | W | 0x0002c000 | Timer Control and Max Function Number Register. |

| Name | Offset | Size | Reset Value | Description |
|--|--------|------|-------------|---|
| <u>PL_SYMBOL_TIMER_FILTER_1_OFF</u> | 0x071c | W | 0x000000280 | Symbol Timer Register and Filter Mask 1 Register. The Filter Mask 1 Register modifies the RADM filtering and error handling rules. For more details, see the "Receive Filtering" section. |
| <u>PL_FILTER_MASK_2_OFF</u> | 0x0720 | W | 0x000000000 | Filter Mask 2 Register. This register modifies the RADM filtering and error handling rules. |
| <u>PL_AMBA_MUL_OB_DECOMPOSED_NP_SUB_REQ_CTRL_OFF</u> | 0x0724 | W | 0x000000001 | AMBA Multiple Outbound Decomposed NP SubRequests Control Register. |
| <u>PL_PL_DEBUG0_OFF</u> | 0x0728 | W | 0x000000000 | Debug Register 0 |
| <u>PL_PL_DEBUG1_OFF</u> | 0x072c | W | 0x000000000 | Debug Register 1 |
| <u>PL_TX_P_FC_CREDIT_STATUS_OFF</u> | 0x0730 | W | 0x000000000 | Transmit Posted FC Credit Status |
| <u>PL_TX_NP_FC_CREDIT_STATUS_OFF</u> | 0x0734 | W | 0x000000000 | TX_NP_FC_CREDIT_STATUS_OFF |
| <u>PL_TX_CPL_FC_CREDIT_STATUS_OFF</u> | 0x0738 | W | 0x000000000 | Transmit Completion FC Credit Status |
| <u>PL_QUEUE_STATUS_OFF</u> | 0x073c | W | 0x000000000 | Queue Status |
| <u>PL_VC_TX_ARBI_1_OFF</u> | 0x0740 | W | 0x0000000f | VC Transmit Arbitration Register 1 |
| <u>PL_VC_TX_ARBI_2_OFF</u> | 0x0744 | W | 0x000000000 | VC Transmit Arbitration Register 2 |
| <u>PL_VCO_P_RX_Q_CTRL_OF</u> | 0x0748 | W | 0x45027048 | Segmented-Buffer VC0 Posted Receive Queue Control. |
| <u>PL_VCO_NP_RX_Q_CTRL_OF</u> | 0x074c | W | 0x05027009 | Segmented-Buffer VC0 Non-Posted Receive Queue Control. |
| <u>PL_VCO_CPL_RX_Q_CTRL_OF</u> | 0x0750 | W | 0x050000000 | Segmented-Buffer VC0 Completion Receive Queue Control. |
| <u>PL_GEN2_CTRL_OF</u> | 0x080c | W | 0x000102ff | Link Width and Speed Change Control Register. |
| <u>PL_PHY_STATUS_OF</u> | 0x0810 | W | 0x000000000 | PHY Status Register. Memory mapped register from phy_cfg_status GPIO input pins. |
| <u>PL_PHY_CONTROL_OF</u> | 0x0814 | W | 0x000000000 | PHY Control Register. Memory mapped register to cfg_phy_control GPIO output pins. |
| <u>PL_TRGT_MAP_CTRL_OF</u> | 0x081c | W | 0x000000047 | Programmable Target Map Control Register. |
| <u>PL_CLOCK_GATING_CTRL_OF</u> | 0x088c | W | 0x000000001 | RADM clock gating enable control register. |
| <u>PL_GEN3 RELATED_OF</u> | 0x0890 | W | 0x00002001 | Gen3 Control Register. This register is reserved for future use. |

| Name | Offset | Size | Reset Value | Description |
|------------------------------------|--------|------|-------------|--|
| PL_GEN3_EQ_CONTROL_OFF | 0x08a8 | W | 0x04059f61 | Gen3 EQ Control Register. |
| PL_GEN3_EQ_FB_MODE_DIR_CHANGE_OFF | 0x08ac | W | 0x00000040 | Gen3 EQ Direction Change Feedback Mode Control Register. |
| PL_ORDER_RULE_CTRL_OFF | 0x08b4 | W | 0x00000000 | Order Rule Control Register. |
| PL_PIPE_LOOPBACK_CONTROL_OFF | 0x08b8 | W | 0x00000003 | PIPE Loopback Control Register. |
| PL_MISC_CONTROL_1_OF_F | 0x08bc | W | 0x00000000 | DBI Read-Only Write Enable Register. |
| PL_MULTI_LANE_CONTROL_OFF | 0x08c0 | W | 0x00000080 | UpConfigure Multi-lane Control Register. |
| PL_PHY_INTEROP_CTRL_OFF | 0x08c4 | W | 0x0000003f | PHY Interoperability Control Register. |
| PL_TRGT_CPL_LUT_DELETE_ENTRY_OFF | 0x08c8 | W | 0x00000000 | TRGT_CPL_LUT Delete Entry Control register. |
| PL_LINK_FLUSH_CONTROL_OFF | 0x08cc | W | 0x00000001 | Link Reset Request Flush Control Register. |
| PL_AMBA_ERROR_RESPONSE_DEFAULT_OFF | 0x08d0 | W | 0x00009c00 | AXI Bridge Slave Error Response Register. |
| PL_AMBA_LINK_TIMEOUT_OFF | 0x08d4 | W | 0x00000032 | Link Down AXI Bridge Slave Timeout Register. |
| PL_AMBA_ORDERING_CTL_OFF | 0x08d8 | W | 0x00000000 | AMBA Ordering Control. |
| PL_COHERENCY_CONTROL_1_OFF | 0x08e0 | W | 0x00000000 | ACE Cache Coherency Control Register 1 |
| PL_COHERENCY_CONTROL_2_OFF | 0x08e4 | W | 0x00000000 | ACE Cache Coherency Control Register 2 |
| PL_COHERENCY_CONTROL_3_OFF | 0x08e8 | W | 0x00000000 | ACE Cache Coherency Control Register 3 |
| PL_AXI_MSTR_MSG_ADDR_LOW_OFF | 0x08f0 | W | 0x00000000 | Lower 20 bits of the programmable AXI address where Messages coming from wire are mapped to. |
| PL_AXI_MSTR_MSG_ADDR_HIGH_OFF | 0x08f4 | W | 0x00000000 | Upper 32 bits of the programmable AXI address where Messages coming from wire are mapped to. |
| PL_PCIE_VERSION_NUMBER_OFF | 0x08f8 | W | 0x3531302a | PCIe Controller IIP Release Version Number. |
| PL_PCIE_VERSION_TYPE_OFF | 0x08fc | W | 0x67612a2a | PCIe Controller IIP Release Version Type. |
| PL_MSIX_ADDRESS_MATCH_LOW_OFF | 0x0940 | W | 0x00000000 | MSI-X Address Match Low Register. |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------|--------|------|-------------|--|
| PL_MSIX_ADDRESS_MAT_CH_HIGH_OFF | 0x0944 | W | 0x00000000 | MSI-X Address Match High Register. |
| PL_MSIX_DOORBELL_OFF | 0x0948 | W | 0x00000000 | MSI-X Doorbell Register. |
| PL_MSIX_RAM_CTRL_OFF | 0x094c | W | 0x00000000 | MSI-X RAM power mode and debug control register. |
| PL_LTR_LATENCY_OFF | 0x0b30 | W | 0x00000000 | LTR Latency Register. |
| PL_AUX_CLK_FREQ_OFF | 0x0b40 | W | 0x00000018 | Auxiliary Clock Frequency Control Register. |
| PL_L1_SUBSTATES_OFF | 0x0b44 | W | 0x000000d2 | L1 Substates Timing Register. |
| PL_PIPE RELATED OFF | 0x0b90 | W | 0x00000000 | PIPE Related Register. |

Notes: **S**-Size: **B**- Byte (8 bits) access, **H****W**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.31 Port Logic Detail Register Description

PL_ACK_LATENCY_TIMER_OFF

Address: Operational Base + offset (0x0700)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x1846 | <p>REPLAY_TIME_LIMIT</p> <p>Replay Timer Limit. The replay timer expires when it reaches this limit. The controller initiates a replay upon reception of a NAK or when the replay timer expires. For more details, see "Transmit Replay". You can modify the effective timer limit with the TIMER_MOD_REPLAY_TIMER field of the TIMER_CTRL_MAX_FUNC_NUM_OFF register. After reset, the controller updates the default according to the Negotiated Link Width, Max_Payload_Size, and speed. The value is determined from Tables 3-4, 3-5, and 3-6 of the PCIe 3.0 specification. If there is a change in the payload size or link speed, the controller will override any value that you have written to this register field, and reset the field back to the specification-defined value. It will not change the value in the TIMER_MOD_REPLAY_TIMER field of the TIMER_CTRL_MAX_FUNC_NUM_OFF register.</p> |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 15:0 | RW | 0x0817 | ROUND_TRIP_LATENCY_TIME_LIMIT Ack Latency Timer Limit. The Ack latency timer expires when it reaches this limit. For more details, see "Ack Scheduling". You can modify the effective timer limit with the TIMER_MOD_ACK_NAK field of the TIMER_CTRL_MAX_FUNC_NUM_OFF register. After reset, the controller updates the default according to the Negotiated Link Width, Max_Payload_Size, and speed. The value is determined from Tables 3-7, 3-8, and 3-9 of the PCIe 3.0 specification. The limit must reflect the round trip latency from requester to completer. If there is a change in the payload size or link width, the controller will override any value that you have written to this register field, and reset the field back to the specification-defined value. It will not change the value in the TIMER_MOD_ACK_NAK field of the TIMER_CTRL_MAX_FUNC_NUM_OFF register. |

PL VENDOR SPEC DLLP OFF

Address: Operational Base + offset (0x0704)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0xffffffff | VENDOR_SPEC_DLLP Vendor Specific DLLP Register. Used to send a specific PCI Express DLLP. Your application writes the 8-bit DLLP Type and 24-bits of Payload data into this register, then sets the field VENDOR_SPECIFIC_DLLP_REQ of PORT_LINK_CTRL_OFF to send the DLLP. [7:0] = Type [31:8] = Payload (24 bits) The dllp type is in bits [7:0] while the remainder is the vendor defined payload. Note: This register field is sticky. |

PL PORT FORCE OFF

Address: Operational Base + offset (0x0708)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | <p>DO_DESKEW_FOR_SRIS Use the transitions from TS2 to Logical Idle Symbol, SKP OS to Logical Idle Symbol, and FTS Sequence to SKP OS to do deskew for SRIS instead of using received SKP OS if DO_DESKEW_FOR_SRIS is set to 1. Note: This register field is sticky.</p> |
| 22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x00 | <p>LINK_STATE Forced LTSSM State. The LTSSM state that the controller is forced to when you set the FORCE_EN bit (Force Link). LTSSM state encoding is defined by the lts_state variable in workspace/src/Layer1/smlh_ltssm.v. Note: This register field is sticky.</p> |
| 15 | WO | 0x0 | <p>FORCE_EN Force Link. The controller supports a testing and debug capability to allow your software to force the LTSSM state machine into a specific state, and to force the controller to transmit a specific Link Command. Asserting this bit triggers the following actions: Forces the LTSSM to the state specified by the Forced LTSSM State field. Forces the controller to transmit the command specified by the Forced Link Command field. This is a self-clearing register field. Reading from this register field always returns a "0".</p> |
| 14:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x0 | <p>FORCED_LTSSM Forced Link Command. The link command that the controller is forced to transmit when you set FORCE_EN bit (Force Link). Link command encoding is defined by the ltssm_cmd variable in workspace/src/Layer1/smlh_ltssm.v. Note: This register field is sticky.</p> |
| 7:0 | RW | 0x04 | <p>LINK_NUM Link Number. Not used for endpoint. Note: This register field is sticky.</p> |

PL ACK F ASPM CTRL OFF

Address: Operational Base + offset (0x070c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 30 | RW | 0x0 | <p>ENTER_ASPM ASPM L1 Entry Control.</p> <p>1: Core enters ASPM L1 after a period in which it has been idle. 0: Core enters ASPM L1 only after idle period during which both receive and transmit are in L0s.</p> <p>Note: This register field is sticky.</p> |
| 29:27 | RW | 0x3 | <p>L1_ENTRANCE_LATENCY L1 Entrance Latency. Value range is:</p> <ul style="list-style-type: none"> 000: 1 us 001: 2 us 010: 4 us 011: 8 us 100: 16 us 101: 32 us 110 or 111: 64 us <p>Note: Programming this timer with a value greater than 32us has no effect unless extended sync is used, or all of the credits are infinite.</p> <p>Note: This register field is sticky.</p> |
| 26:24 | RW | 0x3 | <p>LOS_ENTRANCE_LATENCY L0s Entrance Latency. Values correspond to:</p> <ul style="list-style-type: none"> 000: 1 us 001: 2 us 010: 3 us 011: 4 us 100: 5 us 101: 6 us 110 or 111: 7 us <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 23:16 | RO | 0xff | <p>COMMON_CLK_N_FTS Common Clock N_FTS. This is the N_FTS when common clock is used. The number of Fast Training Sequence ordered sets to be transmitted when transitioning from L0s to L0. The maximum number of FTS ordered-sets that a component can request is 255. This field is only writable (sticky) when all of the following configuration parameter equations are true:</p> <p>CX_NFTS != CX_COMM_NFTS DEFAULT_L0S_EXIT_LATENCY != DEFAULT_COMM_L0S_EXIT_LATENCY DEFAULT_L1_EXIT_LATENCY != DEFAULT_COMM_L1_EXIT_LATENCY</p> <p>The controller does not support a value of zero; a value of zero can cause the LTSSM to go into the recovery state when exiting from L0s.</p> <p>Note: The access attributes of this field are as follows: Wire: R Dbi: R</p> |
| 15:8 | RW | 0xff | <p>ACK_N_FTS N_FTS. The number of Fast Training Sequence ordered sets to be transmitted when transitioning from L0s to L0. The maximum number of FTS ordered-sets that a component can request is 255. The controller does not support a value of zero; a value of zero can cause the LTSSM to go into the recovery state when exiting from L0s. This field is reserved (fixed to '0') for M-PCIe.</p> <p>Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>ACK_FREQ Ack Frequency. The controller accumulates the number of pending ACKs specified here (up to 255) before scheduling an ACK DLLP.</p> <p>0: Indicates that this Ack Frequency Counter feature is turned off. The controller generates a low-priority ACK request for every TLP that it receives. The controller waits until the ACK Latency Timer expires, then converts the current low-priority ACK request to a high-priority ACK request and schedules the DLLP for transmission to the remote link partner.</p> <p>1-255: Indicates that the controller will schedule a high-priority ACK after receiving this number of TLPs. It might schedule the ACK before receiving this number of TLPs if the ACK Latency Timer expires, but never later.</p> <p>For a typical system, you do not have to modify the default setting. For more details, see "ACK/NAK Scheduling".</p> <p>Note: This register field is sticky.</p> |

PL PORT LINK CTRL OFF

Address: Operational Base + offset (0x0710)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27 | RW | 0x0 | TRANSMIT_LANE_REVERSE_ENABLE TRANSMIT_LANE_REVERSE_ENABLE is an internally reserved field. Do not use. Note: This register field is sticky. |
| 26 | RW | 0x0 | EXTENDED_SYNCH EXTENDED_SYNCH is an internally reserved field. Do not use. Note: This register field is sticky. |
| 25 | RW | 0x0 | CORRUPT_LCRC_ENABLE CORRUPT_LCRC_ENABLE is an internally reserved field. Do not use. Note: This register field is sticky. |
| 24 | RW | 0x0 | BEACON_ENABLE BEACON_ENABLE is an internally reserved field. Do not use. Note: This register field is sticky. |
| 23:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x03 | LINK_CAPABLE Link Mode Enable. Sets the number of lanes in the link that you want to connect to the link partner. When you have unused lanes in your system, then you must change the value in this register to reflect the number of lanes. You must also change the value in the "Predetermined Number of Lanes" field of the "Link Width and Speed Change Control Register". For more information, see "How to Tie Off Unused Lanes". For information on upsizing and downsizing the link width, see "Link Establishment". 000001: x1 000011: x2 000111: x4 Note: This register field is sticky. |
| 15:12 | RO | 0x0 | reserved |
| 11:8 | RW | 0x1 | LINK_RATE LINK_RATE is an internally reserved field. Do not use. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 7 | RW | 0x0 | <p>FAST_LINK_MODE Fast Link Mode. Sets all internal LTSSM millisecond timers to Fast Mode for speeding up simulation. Forces the LTSSM training (link initialization) to use shorter time-outs and to link up faster. The default scaling factor can be changed using the DEFAULT_FAST_LINK_SCALING_FACTOR parameter or through the FAST_LINK_SCALING_FACTOR field in the TIMER_CTRL_MAX_FUNC_NUM_OFF register. Fast Link Mode can also be activated by setting the diag_ctrl_bus[2] pin to '1'. For more details, see the "Fast Link Simulation Mode" section in the "Integrating the Core with the PHY or Application RTL or Verification IP" chapter of the User Guide.</p> <p>For M-PCIe, this field also affects Remain Hibern8 Time, Minimum Activate Time, and RRAP timeout. If this bit is set to '1', tRRAPInitiatorResponse is set to 1.88 ms(60 ms/32).</p> <p>Note: This register field is sticky.</p> |
| 6 | RW | 0x0 | <p>LINK_DISABLE LINK_DISABLE is an internally reserved field. Do not use.</p> <p>Note: This register field is sticky.</p> |
| 5 | RW | 0x1 | <p>DLL_LINK_EN DLL Link Enable. Enables link initialization. When DLL Link Enable =0, the controller does not transmit InitFC DLLPs and does not establish a link.</p> <p>Note: This register field is sticky.</p> |
| 4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | <p>RESET_ASSERT Reset Assert. Triggers a recovery and forces the LTSSM to the hot reset state (downstream port only).</p> <p>Note: This register field is sticky.</p> |
| 2 | RW | 0x0 | <p>LOOPBACK_ENABLE Loopback Enable. Turns on loopback. For more details, see "Loopback". For M-PCIe, to force the master to enter Digital Loopback mode, you must set this field to "1" during Configuration.start state(initial discovery/configuration). M-PCIe doesn't support loopback mode from L0 state - only from Configuration.start.</p> <p>Note: This register field is sticky.</p> |
| 1 | RW | 0x0 | <p>SCRAMBLE_DISABLE Scramble Disable. Turns off data scrambling.</p> <p>Note: This register field is sticky.</p> |
| 0 | RW | 0x0 | <p>VENDOR_SPECIFIC_DLLP_REQ Vendor Specific DLLP Request. When software writes a '1' to this bit, the controller transmits the DLLP contained in the VENDOR_SPEC_DLLP field of VENDOR_SPEC_DLLP_OFF. Reading from this self-clearing register field always returns a '0'.</p> |

PL_LANE_SKEW_OFF

Address: Operational Base + offset (0x0714)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RO | 0x0 | reserved |
| 30:27 | RW | 0x1 | <p>IMPLEMENT_NUM_LANES Implementation-specific Number of Lanes. Set the implementation-specific number of lanes. Allowed values are: 4'b0000: 1 lane 4'b0001: 2 lanes 4'b0011: 4 lanes 4'b0111: 8 lanes 4'b1111: 16 lanes</p> <p>The number of lanes to be used when in Loopback Master. The number of lanes programmed must be equal to or less than the valid number of lanes set in LINK_CAPABLE field. You must configure this field before initiating Loopback by writing in the LOOPBACK_ENABLE field. The controller will transition from Loopback.Entry to Loopback.Active after receiving two consecutive TS1 Ordered Sets with the Loopback bit asserted on the implementation specific number of lanes configured in this field.</p> <p>Note: This register field is sticky.</p> |
| 26 | RW | 0x0 | <p>ELASTIC_BUFFER_MODE Selects Elasticity Buffer operating mode: 0: Nominal Half Full Buffer mode 1: Nominal Empty Buffer Mode</p> <p>Note: This register field is sticky.</p> |
| 25 | RW | 0x0 | <p>ACK_NAK_DISABLE Ack/Nak Disable. Prevents the controller from sending ACK and NAK DLLPs.</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>FLOW_CTRL_DISABLE Flow Control Disable. Prevents the controller from sending FC DLLPs.</p> <p>Note: This register field is sticky.</p> |
| 23:0 | RW | 0x000000 | <p>INSERT_LANE_SKEW INSERT_LANE_SKEW is an internally reserved field. Do not use.</p> <p>Note: This register field is sticky.</p> |

PL_TIMER_CTRL_MAX_FUNC_NUM_OFF

Address: Operational Base + offset (0x0718)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 30:29 | RW | 0x0 | <p>FAST_LINK_SCALING_FACTOR Fast Link Timer Scaling Factor. Sets the scaling factor of LTSSM timer when FAST_LINK_MODE field in PORT_LINK_CTRL_OFF is set to '1'. 0: Scaling Factor is 1024 (1ms is 1us *a) 1: Scaling Factor is 256 (1ms is 4us) 2: Scaling Factor is 64 (1ms is 16us) 3: Scaling Factor is 16 (1ms is 64us) Note: This register field is sticky.</p> |
| 28:24 | RW | 0x00 | <p>UPDATE_FREQ_TIMER UPDATE_FREQ_TIMER is an internally reserved field. Do not use. Note: This register field is sticky.</p> |
| 23:19 | RW | 0x00 | <p>TIMER_MOD_ACK_NAK Ack Latency Timer Modifier. Increases the timer value for the Ack latency timer in increments of 64 clock cycles. A value of "0" represents no modification to the timer value. For more details, see the ROUND_TRIP_LATENCY_TIME_LIMIT field of the ACK_LATENCY_TIMER_OFF register. Note: This register field is sticky.</p> |
| 18:14 | RW | 0x0b | <p>TIMER_MOD_REPLY_TIMER Replay Timer Limit Modifier. Increases the time-out value for the replay timer in increments of 64 clock cycles at Gen1 or Gen2 speed, and in increments of 256 clock cycles at Gen3 speed. A value of "0" represents no modification to the timer limit. For more details, see the REPLAY_TIME_LIMIT field of the ACK_LATENCY_TIMER_OFF register. At Gen3 speed, the controller automatically changes the value of this field to DEFAULT_GEN3_REPLY_ADJ. For M-PCIe, this field increases the time-out value for the replay timer in increments of 64 clock cycles at HS-Gear1, HS-Gear2, or HS-Gear3 speed. Note: This register field is sticky.</p> |
| 13:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | <p>MAX_FUNC_NUM Maximum function number that can be used in a request. Configuration requests targeted at function numbers above this value are returned with UR (unsupported request). Note: This register field is sticky.</p> |

PL SYMBOL TIMER FILTER 1 OFF

Address: Operational Base + offset (0x071c)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 31 | RW | 0x0 | <p>CX_FLT_MASK_RC_CFG_DISCARD 0: For RADM RC filter to not allow CFG transaction being received 1: For RADM RC filter to allow CFG transaction being received</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 30 | RW | 0x0 | CX_FLT_MASK_RC_IO_DISCARD 0: For RADM RC filter to not allow IO transaction being received 1: For RADM RC filter to allow IO transaction being received |
| 29 | RW | 0x0 | CX_FLT_MASK_MSG_DROP 0: Drop MSG TLP (except for Vendor MSG). Send decoded message on the SII. 1: Do not Drop MSG (except for Vendor MSG). Send message TLPs to your application on TRGT1 and send decoded message on the SII. The default for this bit is the inverse of FLT_DROP_MSG. That is, if FLT_DROP_MSG =1, then the default of this bit is "0" (drop message TLPs). This bit only controls message TLPs other than Vendor MSGs. Vendor MSGs are controlled by Filter Mask Register 2, bits [1:0]. The controller never passes ATS Invalidate messages to the SII interface regardless of this filter rule setting. The controller passes all ATS Invalidate messages to TRGT1 (or AXI bridge master), as they are too big for the SII. |
| 28 | RW | 0x0 | CX_FLT_MASK_CPL_ECRC_DISCARD Only used when completion queue is advertised with infinite credits and is in store-and-forward mode. 0: Discard completions with ECRC errors 1: Allow completions with ECRC errors to be passed up Reserved field for SW. |
| 27 | RW | 0x0 | CX_FLT_MASK_ECRC_DISCARD 0: Discard TLPs with ECRC errors 1: Allow TLPs with ECRC errors to be passed up |
| 26 | RW | 0x0 | CX_FLT_MASK_CPL_LEN_MATCH 0: Enforce length match for completions; a violation results in cpl_abort, and possibly AER of unexp_cpl_err 1: MASK length match for completions |
| 25 | RW | 0x0 | CX_FLT_MASK_CPL_ATTR_MATCH 0: Enforce attribute match for completions; a violation results in a malformed TLP error, and possibly AER of unexp_cpl_err, cpl_rcvd_ur, cpl_rcvd_ca 1: Mask attribute match for completions |
| 24 | RW | 0x0 | CX_FLT_MASK_CPL_TC_MATCH 0: Enforce Traffic Class match for completions; a violation results in a malformed TLP error, and possibly AER of unexp_cpl_err, cpl_rcvd_ur, cpl_rcvd_ca 1: Mask Traffic Class match for completions |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 23 | RW | 0x0 | CX_FLT_MASK_CPL_FUNC_MATCH 0: Enforce function match for completions; a violation results in cpl_abort, and possibly AER of unexp_cpl_err, cpl_rcvd_ur, cpl_rcvd_ca 1: Mask function match for completions |
| 22 | RW | 0x0 | CX_FLT_MASK_CPL_REQID_MATCH 0: Enforce Req. Id match for completions; a violation result in cpl_abort, and possibly AER of unexp_cpl_err, cpl_rcvd_ur, cpl_rcvd_ca 1: Mask Req. Id match for completions |
| 21 | RW | 0x0 | CX_FLT_MASK_CPL_TAGERR_MATCH 0: Enforce Tag Error Rules for completions; a violation result in cpl_abort, and possibly AER of unexp_cpl_err, cpl_rcvd_ur, cpl_rcvd_ca 1: Mask Tag Error Rules for completions |
| 20 | RW | 0x0 | CX_FLT_MASK_LOCKED_RD_AS_UR 0: Treat locked Read TLPs as UR for EP; Supported for RC 1: Treat locked Read TLPs as Supported for EP; UR for RC |
| 19 | RW | 0x0 | CX_FLT_MASK_CFG_TYPE1_REQ_AS_UR 0: Treat CFG type1 TLPs as UR for EP; Supported for RC 1: Treat CFG type1 TLPs as Supported for EP; UR for RC When CX_SRIOV_ENABLE is set then this bit is set to allow the filter to process Type 1 Config requests if the EP consumes more than one bus number. |
| 18 | RW | 0x0 | CX_FLT_MASK_UR_OUTSIDE_BAR 0: Treat out-of-bar TLPs as UR 1: Do not treat out-of-bar TLPs as UR |
| 17 | RW | 0x0 | CX_FLT_MASK_UR_POIS 0: Treat poisoned request TLPs as UR 1: Do not treat poisoned request TLPs as UR The native controller always passes poisoned completions to your application except when you are using the DMA read channel. |
| 16 | RW | 0x0 | CX_FLT_MASK_UR_FUNC_MISMATCH 0: Treat Function MisMatched TLPs as UR 1: Do not treat Function MisMatched TLPs as UR |
| 15 | RW | 0x0 | DISABLE_FC_WD_TIMER Disable FC Watchdog Timer. Note: This register field is sticky. |
| 14:11 | RW | 0x0 | EIDLE_TIMER EIDLE_TIMER is an internally reserved field. Do not use. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 10:0 | RW | 0x280 | <p>SKP_INT_VAL SKP Interval Value. The number of symbol times to wait between transmitting SKP ordered sets. Note that the controller actually waits the number of symbol times in this register plus 1 between transmitting SKP ordered sets. Your application must program this register accordingly. For example, if 1536 were programmed into this register (in a 250 MHz controller), then the controller actually transmits SKP ordered sets once every 1537 symbol times. The value programmed to this register is actually clock ticks and not symbol times. In a 125 MHz controller, programming the value programmed to this register should be scaled down by a factor of 2 (because one clock tick = two symbol times in this case).</p> <p>Note: This value is not used at Gen3 speed; the skip interval is hardcoded to 370 blocks. For M-PCIe configurations, if the 2K_PPM_DISABLED field in the M-PCIe Configuration Attribute is changed, then this field is changed automatically as follows.</p> <p>2K_PPM_DISABLED=1: 1280 / CX_NB 2K_PPM_DISABLED=0: 228/CX_NB</p> <p>You need to set this field again if necessary when 2K_PPM_DISABLED is changed.</p> <p>Note: This register field is sticky.</p> |

PL FILTER MASK 2 OFF

Address: Operational Base + offset (0x0720)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | CX_FLT_MASK_POIS_ERROR_REPORTING 0: Disable masking of error reporting for Poisoned TLPs 1: Enable masking of error reporting for Poisoned TLPs |
| 7 | RW | 0x0 | CX_FLT_MASK_PRS_DROP 0: Allow PRS message to pass through 1: Drop PRS Messages silently This bit is ignored when the CX_FLT_MASK_MSG_DROP bit in the MASK_RADM_1 field of the SYMBOL_TIMER_FILTER_1_OFF register is set to '1'. |
| 6 | RW | 0x0 | CX_FLT_UNMASK_TD 0: Disable unmask TD bit if CX_STRIP_ECRC_ENABLE 1: Enable unmask TD bit if CX_STRIP_ECRC_ENABLE |
| 5 | RW | 0x0 | CX_FLT_UNMASK_UR_POIS_TRGT0 0: Disable unmask CX_FLT_MASK_UR_POIS with TRGT0 destination 1: Enable unmask CX_FLT_MASK_UR_POIS with TRGT0 destination |
| 4 | RW | 0x0 | CX_FLT_MASK_LN_VENMSG1_DROP 0: Allow LN message to pass through 1: Drop LN Messages silently |
| 3 | RW | 0x0 | CX_FLT_MASK_HANDLE_FLUSH 0: Disable controller Filter to handle flush request 1: Enable controller Filter to handle flush request |
| 2 | RW | 0x0 | CX_FLT_MASK_DABORT_4UCPL 0: Enable DLLP abort for unexpected completion 1: Do not enable DLLP abort for unexpected completion |
| 1 | RW | 0x0 | CX_FLT_MASK_VENMSG1_DROP 0: Vendor MSG Type 1 dropped silently 1: Vendor MSG Type 1 not dropped |
| 0 | RW | 0x0 | CX_FLT_MASK_VENMSG0_DROP 0: Vendor MSG Type 0 dropped with UR error reporting 1: Vendor MSG Type 0 not dropped |

PL_AMBA_MUL_OB_DECOMP_NP_SUB_REQ_CTRL_OFF

Address: Operational Base + offset (0x0724)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RW | 0x1 | <p>OB_RD_SPLIT_BURST_EN Enable AMBA Multiple Outbound Decomposed NP SubRequests. This bit when set to "0" disables the possibility of having multiple outstanding non-posted requests that were derived from decomposition of an outbound AMBA request. For more details, see "AXI Bridge Ordering" in the AXI chapter of the Databook. You should not clear this register unless your application master is requesting an amount of read data greater than Max_Read_Request_Size, and the remote device (or switch) is reordering completions that have different tags.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W (sticky) Dbi: R/W (sticky)</p> <p>Note: This register field is sticky.</p> |

PL PL DEBUG0 OFF

Address: Operational Base + offset (0x0728)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RO | 0x00000000 | DEB_REG_0 The value on cxpl_debug_info[31:0]. |

PL PL DEBUG1 OFF

Address: Operational Base + offset (0x072c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | DEB_REG_1 The value on cxpl_debug_info[63:32]. |

PL TX P FC CREDIT STATUS OFF

Address: Operational Base + offset (0x0730)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:12 | RO | 0x00 | TX_P_HEADER_FC_CREDIT Transmit Posted Header FC Credits. The posted Header credits advertised by the receiver at the other end of the link, updated with each UpdateFC DLLP. Default value depends on the number of advertised credits for header and data [12'b0, xtlh_xadm_ph_cdts, xtlh_xadm_pd_cdts]; When the number of advertised completion credits (both header and data) are infinite, then the default would be [12'b0, 8'hFF, 12'hFFF]. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11:0 | RO | 0x000 | TX_P_DATA_FC_CREDIT Transmit Posted Data FC Credits. The posted Data credits advertised by the receiver at the other end of the link, updated with each UpdateFC DLLP. Default value depends on the number of advertised credits for header and data [12'b0, xtlh_xadm_ph_cdts, xtlh_xadm_pd_cdts]; When the number of advertised completion credits (both header and data) are infinite, then the default would be [12'b0, 8'hFF, 12'hFFF]. |

PL TX NP FC CREDIT STATUS OFF

Address: Operational Base + offset (0x0734)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x0 | reserved |
| 19:12 | RO | 0x00 | TX_NP_HEADER_FC_CREDIT Transmit Non-Posted Header FC Credits. The non-posted Header credits advertised by the receiver at the other end of the link, updated with each UpdateFC DLLP. Default value depends on the number of advertised credits for header and data [12'b0, xtlh_xadm_nph_cdts, xtlh_xadm_npd_cdts]; When the number of advertised completion credits (both header and data) are infinite, then the default would be [12'b0, 8'hFF, 12'hFFF]. |
| 11:0 | RO | 0x000 | TX_NP_DATA_FC_CREDIT Transmit Non-Posted Data FC Credits. The non-posted Data credits advertised by the receiver at the other end of the link, updated with each UpdateFC DLLP. Default value depends on the number of advertised credits for header and data [12'b0, xtlh_xadm_nph_cdts, xtlh_xadm_npd_cdts]; When the number of advertised completion credits (both header and data) are infinite, then the default would be [12'b0, 8'hFF, 12'hFFF]. |

PL TX CPL FC CREDIT STATUS OFF

Address: Operational Base + offset (0x0738)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | reserved |
| 19:12 | RO | 0x00 | TX_CPL_HEADER_FC_CREDIT Transmit Completion Header FC Credits. The Completion Header credits advertised by the receiver at the other end of the link, updated with each UpdateFC DLLP. Default value depends on the number of advertised credits for header and data [12'b0, xtlh_xadm_cplh_cdts, xtlh_xadm_cpld_cdts]; When the number of advertised completion credits (both header and data) are infinite, then the default would be [12'b0, 8'hFF, 12'hFFF]. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11:0 | RO | 0x000 | TX_CPL_DATA_FC_CREDIT Transmit Completion Data FC Credits. The Completion Data credits advertised by the receiver at the other end of the link, updated with each UpdateFC DLLP. Default value depends on the number of advertised credits for header and data [12'b0, xtlh_xadm_cplh_cdts, xtlh_xadm_cpld_cdts]; When the number of advertised completion credits (both header and data) are infinite, then the default would be [12'b0, 8'hFF, 12'hFFF]. |

PL_QUEUE_STATUS OFF

Address: Operational Base + offset (0x073c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | TIMER_MOD_FLOW_CONTROL_EN FC Latency Timer Override Enable. When this bit is set, the value from the "FC Latency Timer Override Value" field in this register will override the FC latency timer value that the controller calculates according to the PCIe specification. Note: This register field is sticky. |
| 30:29 | RO | 0x0 | reserved |
| 28:16 | RW | 0x0000 | TIMER_MOD_FLOW_CONTROL FC Latency Timer Override Value. When you set the "FC Latency Timer Override Enable" in this register, the value in this field will override the FC latency timer value that the controller calculates according to the PCIe specification. For more details, see "Flow Control". Note: This register field is sticky. |
| 15:14 | RO | 0x0 | reserved |
| 13 | RO | 0x0 | RX_SERIALIZATION_Q_NON_EMPTY Receive Serialization Queue Not Empty. Indicates there is data in the serialization queue. |
| 12:4 | RO | 0x0 | reserved |
| 3 | RW | 0x0 | RX_QUEUE_OVERFLOW Receive Credit Queue Overflow. Indicates insufficient buffer space available to write to the P/NP/CPL credit queue. |
| 2 | RO | 0x0 | RX_QUEUE_NON_EMPTY Receive Credit Queue Not Empty. Indicates there is data in one or more of the receive buffers. |
| 1 | RO | 0x0 | TX_RETRY_BUFFER_NE Transmit Retry Buffer Not Empty. Indicates that there is data in the transmit retry buffer. |
| 0 | RO | 0x0 | RX_TLP_FC_CREDIT_NON_RETURN Received TLP FC Credits Not Returned. Indicates that the controller has received a TLP but has not yet sent an UpdateFC DLLP indicating that the credits for that TLP have been restored by the receiver at the other end of the link. |

PL VC TX ARBI 1 OFF

Address: Operational Base + offset (0x0740)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x00 | WRR_WEIGHT_VC_3 WRR Weight for VC3. |
| 23:16 | RW | 0x00 | WRR_WEIGHT_VC_2 WRR Weight for VC2. |
| 15:8 | RW | 0x00 | WRR_WEIGHT_VC_1 WRR Weight for VC1. |
| 7:0 | RW | 0x0f | WRR_WEIGHT_VC_0 WRR Weight for VC0. |

PL VC TX ARBI 2 OFF

Address: Operational Base + offset (0x0744)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RW | 0x00 | WRR_WEIGHT_VC_7 WRR Weight for VC7. |
| 23:16 | RW | 0x00 | WRR_WEIGHT_VC_6 WRR Weight for VC6. |
| 15:8 | RW | 0x00 | WRR_WEIGHT_VC_5 WRR Weight for VC5. |
| 7:0 | RW | 0x00 | WRR_WEIGHT_VC_4 WRR Weight for VC4. |

PL VC0 P RX Q CTRL OFF

Address: Operational Base + offset (0x0748)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | VC_ORDERING_RX_Q VC Ordering for Receive Queues. Determines the VC ordering rule for the receive queues, used only in the segmented-buffer configuration: 1: Strict ordering, higher numbered VCs have higher priority 0: Round robin Note: This register field is sticky. |
| 30 | RW | 0x1 | TLP_TYPE_ORDERING_VC0 TLP Type Ordering for VC0. Determines the TLP type ordering rule for VC0 receive queues, used only in the segmented-buffer configuration: 1: PCIe ordering rules (recommended) 0: Strict ordering: posted, completion, then non-posted Note: This register field is sticky. |
| 29:28 | RO | 0x0 | reserved |
| 27:26 | RW | 0x1 | VC0_P_DATA_SCALE VC0 Scale Posted Data Credits. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 25:24 | RW | 0x1 | VCO_P_HDR_SCALE VCO Scale Posted Header Credits. Note: This register field is sticky. |
| 23:20 | RO | 0x0 | reserved |
| 19:12 | RO | 0x27 | VCO_P_HEADER_CREDIT VCO Posted Header Credits. The number of initial posted header credits for VC0, used only in the segmented-buffer configuration. Note: This register field is sticky. |
| 11:0 | RO | 0x048 | VCO_P_DATA_CREDIT VCO Posted Data Credits. The number of initial posted data credits for VC0, used only in the segmented-buffer configuration. Note: This register field is sticky. |

PL_VCO_NP_RX_Q_CTRL_OFF

Address: Operational Base + offset (0x074c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27:26 | RW | 0x1 | VCO_NP_DATA_SCALE VCO Scale Non-Posted Data Credits. Note: This register field is sticky. |
| 25:24 | RW | 0x1 | VCO_NP_HDR_SCALE VCO Scale Non-Posted Header Credits. Note: This register field is sticky. |
| 23:20 | RO | 0x0 | reserved |
| 19:12 | RO | 0x27 | VCO_NP_HEADER_CREDIT VCO Non-Posted Header Credits. The number of initial non-posted header credits for VC0, used only in the segmented-buffer configuration. Note: This register field is sticky. |
| 11:0 | RO | 0x009 | VCO_NP_DATA_CREDIT VCO Non-Posted Data Credits. The number of initial non-posted data credits for VC0, used only in the segmented-buffer configuration. Note: This register field is sticky. |

PL_VCO_CPL_RX_Q_CTRL_OFF

Address: Operational Base + offset (0x0750)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:28 | RO | 0x0 | reserved |
| 27:26 | RW | 0x1 | VCO_CPL_DATA_SCALE VCO Scale CPL Data Credits. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 25:24 | RW | 0x1 | VCO_CPL_HDR_SCALE VCO Scale CPL Header Credits. Note: This register field is sticky. |
| 23:20 | RO | 0x0 | reserved |
| 19:12 | RO | 0x00 | VCO_CPL_HEADER_CREDIT VCO Completion Header Credits. The number of initial Completion header credits for VC0, used only in the segmented-buffer configuration. Note: This register field is sticky. |
| 11:0 | RO | 0x000 | VCO_CPL_DATA_CREDIT VCO Completion Data Credits. The number of initial Completion data credits for VC0, used only in the segmented-buffer configuration. Note: This register field is sticky. |

PL_GEN2_CTRL OFF

Address: Operational Base + offset (0x080c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:22 | RO | 0x0 | reserved |
| 21 | RW | 0x0 | Electrical Idle Inference Mode at Gen1 Rate. Programmable mode to determine inferred electrical idle (EI) in Recovery.Speed or Loopback.Active (as slave) state at Gen1 speed by looking for a "1" value on RxElecIdle instead of looking for a "0" on RxValid. If the PHY fails to deassert the RxValid signal in Recovery.Speed or Loopback.Active (because of corrupted EIOS for example), then EI cannot be inferred successfully in the controller by just detecting the condition RxValid=0. 0: Use RxElecIdle signal to infer Electrical Idle 1: Use RxValid signal to infer Electrical Idle Note: This register field is sticky. |
| 20 | RW | 0x0 | SEL_DEEMPHASIS Used to set the de-emphasis level for upstream ports. This bit selects the level of de-emphasis the link operates at. 0: -6 dB 1: -3.5 dB Note: This register field is sticky. |
| 19 | RW | 0x0 | CONFIG_TX_COMP_RX Config Tx Compliance Receive Bit. When set to 1, signals LTSSM to transmit TS ordered sets with the compliance receive bit assert (equal to "1"). Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 18 | RW | 0x0 | <p>CONFIG_PHY_TX_CHANGE Config PHY Tx Swing. Controls the PHY transmitter voltage swing level. The controller drives the mac_phy_txswing output from this register bit field.</p> <p>0: Full Swing 1: Low Swing</p> <p>Note: This register field is sticky.</p> |
| 17 | RW | 0x0 | <p>DIRECT_SPEED_CHANGE Directed Speed Change. Writing "1" to this field instructs the LTSSM to initiate a speed change to Gen2 or Gen3 after the link is initialized at Gen1 speed. When the speed change occurs, the controller will clear the contents of this field; and a read to this field by your software will return a "0". To manually initiate the speed change:</p> <p>Write to LINK_CONTROL2_LINK_STATUS2_REG . PCIE_CAP_TARGET_LINK_SPEED in the local device Deassert this field Assert this field</p> |
| 16 | RW | 0x1 | <p>AUTO_LANE_FLIP_CTRL_EN Enable Auto flipping of the lanes. You must set the CX_AUTO_LANE_FLIP_CTRL_EN configuration parameter to include the hardware for this feature in the controller.</p> <p>Note: This register field is sticky.</p> |
| 15:13 | RW | 0x0 | <p>PRE_DET_LANE Predetermined Lane for Auto Flip. This field defines which physical lane is connected to logical Lane0 by the flip operation performed in Detect. Allowed values are:</p> <p>3'b000: Connect logical Lane0 to physical lane 0 or CX_NL-1 or CX_NL/2-1 or CX_NL/4-1 or CX_NL/8-1, depending on which lane is detected 3'b001: Connect logical Lane0 to physical lane 1 3'b010: Connect logical Lane0 to physical lane 3 3'b011: Connect logical Lane0 to physical lane 7 3'b100: Connect logical Lane0 to physical lane 15</p> <p>This field is used to restrict the receiver detect procedure to a particular lane when the default detect and polling procedure performed on all lanes cannot be successful. A notable example of when it is useful to program this field to a value different from the default, is when a lane is asymmetrically broken, that is, it is detected in Detect LTSSM state but it cannot exit Electrical Idle in Polling LTSSM state.</p> |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 12:8 | RW | 0x02 | <p>NUM_OF_LANES Predetermined Number of Lanes. Defines the number of lanes which are connected and not bad. Used to limit the effective link width to ignore "broken" or "unused" lanes that detect a receiver. Indicates the number of lanes to check for exit from Electrical Idle in Polling.Active and L2.Idle. It is possible that the LTSSM might detect a receiver on a bad or broken lane during the Detect Substate. However, it is also possible that such a lane might also fail to exit Electrical Idle and therefore prevent a valid link from being configured. This value is referred to as the "Predetermined Number of Lanes" in section 4.2.6.2.1 of the PCI Express Base 3.0 Specification, revision 1.0. Encoding is as follows:</p> <ul style="list-style-type: none"> 0x01: 1 lane 0x02: 2 lanes 0x03: 3 lanes .. <p>When you have unused lanes in your system, then you must change the value in this register to reflect the number of lanes. You must also change the value in the "Link Mode Enable" field of PORT_LINK_CTRL_OFF. The value in this register is normally the same as the encoded value in PORT_LINK_CTRL_OFF. If you find that one of your used lanes is bad then you must reduce the value in this register. For more information, see "How to Tie Off Unused Lanes." For information on upsizing and downsizing the link width, see "Link Establishment."</p> <p>Note: This register field is sticky.</p> |
| 7:0 | RW | 0xff | <p>FAST_TRAINING_SEQ Sets the Number of Fast Training Sequences (N_FTS) that the controller advertises as its N_FTS during Gen2 or Gen3 link training. This value is used to inform the link partner about the PHY's ability to recover synchronization after a low power state. The number should be provided by the PHY vendor. Do not set N_FTS to zero; doing so can cause the LTSSM to go into the recovery state when exiting from L0s. This field is reserved (fixed to '0') for M-PCIe.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Note: This register field is sticky.</p> |

PL PHY STATUS OFF

Address: Operational Base + offset (0x0810)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RO | 0x00000000 | PHY_STATUS PHY Status. Data received directly from the phy_cfg_status bus. These is a GPIO register reflecting the values on the static phy_cfg_status input signals. The usage is left completely to the user and does not in any way influence controller functionality. You can use it for any static sideband status signalling requirements that you have for your PHY. Note: This register field is sticky. |

PL PHY CONTROL OFF

Address: Operational Base + offset (0x0814)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | PHY_CONTROL PHY Control. Data sent directly to the cfg_phy_control bus. These is a GPIO register driving the values on the static cfg_phy_control output signals. The usage is left completely to the user and does not in any way influence controller functionality. You can use it for any static sideband control signalling requirements that you have for your PHY. |

PL TRGT MAP CTRL OFF

Address: Operational Base + offset (0x081c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:21 | RO | 0x0 | reserved |
| 20:16 | RW | 0x00 | TARGET_MAP_INDEX The number of the PF Function on which the Target Values are set. This register does not respect the Byte Enable setting. any write will affect all register bits. |
| 15:7 | RO | 0x0 | reserved |
| 6 | RW | 0x1 | TARGET_MAP_ROM Target Value for the ROM page of the PF Function selected by the index number. This register does not respect the Byte Enable setting. any write will affect all register bits. |
| 5:0 | RW | 0x07 | TARGET_MAP_PF Target Values for each BAR on the PF Function selected by the index number. This register does not respect the Byte Enable setting. any write will affect all register bits. |

PL CLOCK GATING CTRL OFF

Address: Operational Base + offset (0x088c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 0 | RW | 0x1 | RADM_CLK_GATING_EN Enable Radm clock gating feature. 0: Disable 1: Enable |

PL_GEN3 RELATED OFF

Address: Operational Base + offset (0x0890)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | GEN3_EQ_INVREQ_EVAL_DIFF_DISABLE Eq InvalidRequest and RxEqEval Different Time Assertion Disable. Disable the assertion of Eq InvalidRequest and RxEqEval at different time. |
| 22:19 | RO | 0x0 | reserved |
| 18 | RW | 0x0 | GEN3_DC_BALANCE_DISABLE DC Balance Disable. Disable DC Balance feature. |
| 17 | RW | 0x0 | GEN3_DLLP_XMT_DELAY_DISABLE DLLP Transmission Delay Disable. Disable delay transmission of DLLPs before equalization. |
| 16 | RW | 0x0 | GEN3_EQUALIZATION_DISABLE Equalization Disable. Disable equalization feature. This bit cannot be changed once the LTSSM starts link training. |
| 15:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | RXEQ_RGRDLESS_RXTS When set to '1', the controller as Gen3 EQ master asserts RxEqEval to instruct the PHY to do Rx adaptation and evaluation after a 500ns timeout from a new preset request. 0: mac_phy_rxeqeval asserts after 1us and 2 TS1 received from remote partner. 1: mac_phy_rxeqeval asserts after 500ns regardless of TS's received or not. |
| 12 | RW | 0x0 | RXEQ_PH01_EN Rx Equalization Phase 0/Phase 1 Hold Enable. When this bit is set the upstream port holds phase 0 (the downstream port holds phase 1) for 10ms. Holding phase 0 or phase 1 can be used to allow sufficient time for Rx Equalization to be performed by the PHY. This bit is used during Virtex-7 Gen3 equalization. The programmable bits [RXEQ_PH01_EN, EQ_PHASE_2_3] can be used to obtain the following variations of the equalization procedure: 00: Tx equalization only in phase 2/3 01: No Tx equalization, no Rx equalization 10: Tx equalization in phase 2/3, Rx equalization in phase 0/1 11: No Tx equalization, Rx equalization in phase 0/1 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 11 | RW | 0x0 | <p>EQ_RED0 Equalization Redo Disable. Disable autonomous mechanism for requesting to redo the equalization process.</p> <p>Note: This register field is sticky.</p> |
| 10 | RW | 0x0 | <p>EQ_EIEOS_CNT Equalization EIEOS Count Reset Disable. Disable requesting reset of EIEOS count during equalization.</p> <p>Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>EQ_PHASE_2_3 Equalization Phase 2 and Phase 3 Disable. This applies to downstream ports only.</p> |
| 8 | RW | 0x0 | <p>DISABLE_SCRAMBLER_GEN_3 Disable Scrambler for Gen3 and Gen4 Data Rate.</p> |
| 7:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | <p>GEN3_ZRXDC_NONCOMPL Gen3 Receiver Impedance ZRX-DC Not Compliant. Receivers that operate at 8.0 GT/s with an impedance other than the range defined by the ZRX-DC parameter for 2.5 GT/s (40-60 Ohms) must meet additional behavior requirements in the following LTSSM states: Polling, Rx_L0s, L1, L2, and Disabled.</p> <p>0: The receiver complies with the ZRX-DC parameter for 2.5 GT/s when operating at 8 GT/s or higher.</p> <p>1: The receiver does not comply with the ZRX-DC parameter for 2.5 GT/s when operating at 8 GT/s or higher.</p> <p>Note: This register field is sticky.</p> |

PL_GEN3_EQ_CONTROL_OFF

Address: Operational Base + offset (0x08a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x0 | reserved |
| 26 | RW | 0x1 | <p>GEN3_REQ_SEND_CONSEC_EIEOS_FOR_PSET_MAP Request controller to send back-to-back EIEOS in Recovery.RcvrLock state until presets to coefficients mapping is complete.</p> <p>0: Do not request 1: request</p> |
| 25 | RW | 0x0 | <p>GEN3_EQ_PSET_REQ_AS_COEF GEN3_EQ_PSET_REQ_AS_COEF is an internally reserved field. Do not use.</p> |
| 24 | RW | 0x0 | <p>GEN3_EQ_FOM_INC_INITIAL_EVAL Include Initial FOM. Include or not the FOM feedback from the initial preset evaluation performed in the EQ Master, when finding the highest FOM among all preset evaluations.</p> <p>0: Do not include 1: Include</p> |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 23:8 | RW | 0x059f | <p>GEN3_EQ_PSET_REQ_VEC Preset Request Vector. Requesting of Presets during the initial part of the EQ Master Phase. Encoding scheme is as follows:</p> <ul style="list-style-type: none"> Bit [15:0] =0x0: No preset is requested and evaluated in EQ Master Phase. Bit [i] =1: "Preset=i" is requested and evaluated in EQ Master Phase. 0000000000000000: No preset be requested and evaluated in EQ Master Phase 00000xxxxxx1: Preset 0 is requested and evaluated in EQ Master Phase 00000xxxxxxxx1x: Preset 1 is requested and evaluated in EQ Master Phase 00000xxxxxxxx1xx: Preset 2 is requested and evaluated in EQ Master Phase 00000xxxxxxxx1xxx: Preset 3 is requested and evaluated in EQ Master Phase 00000xxxxxxxx1xxxx: Preset 4 is requested and evaluated in EQ Master Phase 00000xxxxxxxx1xxxxx: Preset 5 is requested and evaluated in EQ Master Phase 00000xxx1xxxxxx: Preset 6 is requested and evaluated in EQ Master Phase 00000xx1xxxxxxxx: Preset 7 is requested and evaluated in EQ Master Phase 00000x1xxxxxxxxxx: Preset 8 is requested and evaluated in EQ Master Phase 00000x1xxxxxxxxxxxx: Preset 9 is requested and evaluated in EQ Master Phase 000001xxxxxxxxxxxx: Preset 10 is requested and evaluated in EQ Master Phase All other encodings: Reserved |
| 7 | RO | 0x0 | reserved |
| 6 | RW | 0x1 | <p>GEN3_LOWER_RATE_EQ_REDO_ENABLE Support EQ redo and lower rate change: 0: not support 1: support</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 5 | RW | 0x1 | <p>GEN3_EQ_EVAL_2MS_DISABLE Phase2_3 2 ms Timeout Disable. Determine behavior in Phase2 for USP (Phase3 if DSP) when the PHY does not respond within 2ms to the assertion of RxEqEval: 0: abort the current evaluation, stop any attempt to modify the remote transmitter settings, Phase2 is terminated by the 24ms timeout 1: ignore the 2ms timeout and continue as normal. This is used to support PHYs that require more than 2ms to respond to the assertion of RxEqEval.</p> |
| 4 | RW | 0x0 | <p>GEN3_EQ_PHASE23_EXIT_MODE Behavior After 24 ms Timeout (when optimal settings are not found). For a USP: Determine next LTSSM state from Phase2 after 24ms Timeout 0: Recovery.Speed 1: Recovery.Equalization.Phase3 When optimal settings are not found then: Equalization Phase 2 Successful status bit is not set in the "Link Status Register 2" when GEN3_EQ_PHASE23_EXIT_MODE = 0 Equalization Phase 2 Successful status bit is set in the "Link Status Register 2" when GEN3_EQ_PHASE23_EXIT_MODE = 1 Equalization Phase 2 Complete status bit is set in the "Link Status Register 2" For a DSP: Determine next LTSSM state from Phase3 after 24ms Timeout 0: Recovery.Speed 1: Recovery.Equalization.RcvrLock When optimal settings are not found then: Equalization Phase 3 Successful status bit is not set in the "Link Status Register 2" when GEN3_EQ_PHASE23_EXIT_MODE = 0 Equalization Phase 3 Successful status bit is set in the "Link Status Register 2" when GEN3_EQ_PHASE23_EXIT_MODE = 1 Equalization Phase 3 Complete status bit is set in the "Link Status Register 2"</p> |
| 3:0 | RW | 0x1 | <p>GEN3_EQ_FB_MODE Feedback Mode. 0000b: Direction Change 0001b: Figure Of Merit 0010b: Reserved: Reserved 1111b: Reserved</p> |

PL_GEN3_EQ_FB_MODE_DIR_CHANGE_OFF

Address: Operational Base + offset (0x08ac)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:18 | RO | 0x0 | reserved |
| 17:14 | RW | 0x0 | GEN3_EQ_FMDC_MAX_POST_CUSROR_DELTA Convergence Window Aperture for C+1. Post-cursor coefficients maximum delta within the convergence window depth. Allowed range: 0,1,2,..15. |
| 13:10 | RW | 0x0 | GEN3_EQ_FMDC_MAX_PRE_CUSROR_DELTA Convergence Window Aperture for C-1. Pre-cursor coefficients maximum delta within the convergence window depth. Allowed range: 0,1,2,..15. |
| 9:5 | RW | 0x02 | GEN3_EQ_FMDC_N_EVALS Convergence Window Depth. Number of consecutive evaluations considered in Phase 2/3 when determining if optimal coefficients have been found. Allowed range: 0,1,2,..16 up to a maximum of CX_GEN3_EQ_COEFQ_DEPTH. When set to 0, EQ Master is performed without sending any requests to the remote partner in Phase 2 for USP and Phase 3 for DSP. Therefore, the remote partner will not change its transmitter coefficients and will move to the next state. |
| 4:0 | RW | 0x00 | GEN3_EQ_FMDC_T_MIN_PHASE23 Minimum Time (in ms) To Remain in EQ Master Phase. The LTSSM stays in EQ Master phase for at least this amount of time, before starting to check for convergence of the coefficients. Allowed values 0,1,...,24. |

PL ORDER RULE CTRL OFF

Address: Operational Base + offset (0x08b4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | reserved |
| 15:8 | RW | 0x00 | CPL_PASS_P Completion Passing Posted Ordering Rule Control. Determines if CPL can pass halted P queue. 0: CPL can not pass P (recommended) 1: CPL can pass P |
| 7:0 | RW | 0x00 | NP_PASS_P Non-Posted Passing Posted Ordering Rule Control. Determines if NP can pass halted P queue. 0 : NP can not pass P (recommended). 1 : NP can pass P |

PL PIPE LOOPBACK CONTROL OFF

Address: Operational Base + offset (0x08b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | PIPE_LOOPBACK PIPE Loopback Enable. Indicates RMMI Loopback if M-PCIe. Note: This register field is sticky. |
| 30:27 | RO | 0x0 | reserved |
| 26:24 | RW | 0x0 | RXSTATUS_VALUE RXSTATUS_VALUE is an internally reserved field. Do not use. |
| 23:22 | RO | 0x0 | reserved |
| 21:16 | RW | 0x00 | RXSTATUS_LANE RXSTATUS_LANE is an internally reserved field. Do not use. Note: This register field is sticky. |
| 15:0 | RW | 0x0003 | LPBK_RXVALID LPBK_RXVALID is an internally reserved field. Do not use. Note: This register field is sticky. |

PL_MISC_CONTROL_1 OFF

Address: Operational Base + offset (0x08bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | reserved |
| 5 | RW | 0x0 | ARI_DEVICE_NUMBER When ARI is enabled, this field enables use of the device ID. Note: This register field is sticky. |
| 4 | RW | 0x0 | DISABLE_AUTO_LTR_CLR_MSG Disable the autonomous generation of LTR clear message in upstream port. 0: Allow the autonomous generation of LTR clear message. 1: Disable the autonomous generation of LTR clear message. Default value is 0. For more details, see "Latency Tolerance Reporting (LTR) Message Generation [EP Mode]" in "Message Generation" section of the "Controller Operations" chapter of the Databook. Note: This register field is sticky. |
| 3 | RW | 0x0 | SIMPLIFIED_REPLY_TIMER Enables Simplified Replay Timer (Gen4). For more details, see "Transmit Replay" in the Controller Operations chapter of the Databook. Simplified Replay Timer Values are: A value from 24,000 to 31,000 Symbol Times when Extended Synch is 0b. A value from 80,000 to 100,000 Symbol Times when Extended Synch is 1b. Must not be changed while link is in use. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 2 | RW | 0x0 | <p>UR_CA_MASK_4_TRGT1 This field only applies to request TLPs (with UR filtering status) that you have chosen to forward to the application (when you set DEFAULT_TARGET in this register). When you set this field to '1', the core suppresses error logging, Error Message generation, and CPL generation (for non-posted requests). For more details, refer to the "Advanced Error Handling For Received TLPs" chapter of the Databook. You should set this if you have set the Default Target port logic register to '1'. Note: This register field is sticky.</p> |
| 1 | RW | 0x0 | <p>DEFAULT_TARGET Default target a received IO or MEM request with UR/CA/CRS is sent to by the controller. 0: The controller drops all incoming I/O or MEM requests (after corresponding error reporting). A completion with UR status will be generated for non-posted requests. 1: The controller forwards all incoming I/O or MEM requests with UR/CA/CRS status to your application For more details, see "ECRC Handling" and "Request TLP Routing Rules" in "Receive Routing" section of the "Controller Operations" chapter of the Databook. Default value is DEFAULT_TARGET configuration parameter. Note: This register field is sticky.</p> |
| 0 | RW | 0x0 | <p>DBI_RO_WR_EN Write to RO Registers Using DBI. When you set this field to "1", then some RO and HwInit bits are writable from the local application through the DBI. For more details, see "Writing to Read-Only Registers." Note: This register field is sticky.</p> |

PL MULTI LANE CONTROL OFF

Address: Operational Base + offset (0x08c0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7 | RW | 0x1 | <p>UPCONFIGURE_SUPPORT Upconfigure Support. The controller sends this value as the Link Upconfigure Capability in TS2 Ordered Sets in Configuration.Complete state.</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 6 | RW | 0x0 | <p>DIRECT_LINK_WIDTH_CHANGE Directed Link Width Change. The controller always moves to Configuration state through Recovery state when this bit is set to '1'. If the upconfigure_capable variable is '1' and the PCIE_CAP_HW_AUTO_WIDTH_DISABLE bit in LINK_CONTROL_LINK_STATUS_REG is '0', the controller starts upconfigure or autonomous width downsizing (to the TARGET_LINK_WIDTH value) in the Configuration state. If TARGET_LINK_WIDTH value is 0x0, the controller does not start upconfigure or autonomous width downsizing in the Configuration state. The controller self-clears this field when the controller accepts this request.</p> |
| 5:0 | RW | 0x00 | <p>TARGET_LINK_WIDTH Target Link Width. Values correspond to: 6'b000000: Core does not start upconfigure or autonomous width downsizing in the Configuration state. 6'b000001: x1 6'b000010: x2 6'b000100: x4 6'b001000: x8 6'b010000: x16 6'b100000: x32</p> |

PL_PHY_INTEROP_CTRL_OFF

Address: Operational Base + offset (0x08c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | reserved |
| 10 | RW | 0x0 | <p>L1_CLK_SEL L1 Clock control bit. 1: Controller does not request aux_clk switch and core_clk gating in L1. 0: Controller requests aux_clk switch and core_clk gating in L1. Note: This register field is sticky.</p> |
| 9 | RW | 0x0 | <p>L1_NOWAIT_P1 L1 entry control bit. 1: Core does not wait for PHY to acknowledge transition to P1 before entering L1. 0: Core waits for the PHY to acknowledge transition to P1 before entering L1. Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 8 | RW | 0x0 | <p>L1SUB_EXIT_MODE L1 Exit Control Using phy_mac_pclkack_n. 1: Core exits L1 without waiting for the PHY to assert phy_mac_pclkack_n. 0: Core waits for the PHY to assert phy_mac_pclkack_n before exiting L1. Note: This register field is sticky.</p> |
| 7 | RO | 0x0 | reserved |
| 6:0 | RW | 0x3f | <p>RXSTANDBY_CONTROL Rxstandby Control. Bits 0..5 determine if the controller asserts the RxStandby signal (mac_phy_rxstandby) in the indicated condition. Bit 6 enables the controller to perform the RxStandby/RxStandbyStatus handshake. [0]: Rx EIOS and subsequent T TX-IDLE-MIN [1]: Rate Change [2]: Inactive lane for upconfigure/downconfigure [3]: PowerDown=P1orP2 [4]: RxL0s.Idle [5]: EI Infer in L0 [6]: Execute RxStandby/RxStandbyStatus Handshake Note: This register field is sticky.</p> |

PL TRGT CPL LUT DELETE ENTRY OFF

Address: Operational Base + offset (0x08c8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31 | RO | 0x0 | reserved |
| 30:0 | RW | 0x00000000 | <p>LOOK_UP_ID This number selects one entry to delete of the TRGT_CPL_LUT.</p> |

PL LINK FLUSH CONTROL OFF

Address: Operational Base + offset (0x08cc)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RW | 0x1 | <p>AUTO_FLUSH_EN Enables automatic flushing of pending requests before sending the reset request to the application logic to reset the PCIe controller and the AXI Bridge. The flushing process is initiated if any of the following events occur:</p> <ul style="list-style-type: none"> Hot reset request. A downstream port (DSP) can "hot reset" an upstream port (USP) by sending two consecutive TS1 ordered sets with the hot reset bit asserted. Warm (Soft) reset request. Generated when exiting from D3 to D0 and cfg_pm_no_soft_rst=0. Link down reset request. A high to low transition on smlh_req_rst_not indicates the link has gone down and the controller is requesting a reset. <p>If you disable automatic flushing, your application is responsible for resetting the PCIe controller and the AXI Bridge.</p> <p>Note: This register field is sticky.</p> |

PL AMBA ERROR RESPONSE DEFAULT OFF

Address: Operational Base + offset (0x08d0)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RO | 0x0 | reserved |
| 15:10 | RW | 0x27 | <p>AMBA_ERROR_RESPONSE_MAP AXI Slave Response Error Map. Allows you to selectively map the errors received from the PCIe completion (for non-posted requests) to the AXI slave responses, slv_rresp or slv_bresp. The recommended setting is SLVERR. CRS is always mapped to OKAY.</p> <p>bit 0: 0: UR (unsupported request) -> DECERR 1: UR (unsupported request) -> SLVERR</p> <p>bit 1: 0: CRS (configuration retry status) -> DECERR 1: CRS (configuration retry status) -> SLVERR</p> <p>bit 2 0: CA (completer abort) -> DECERR 1: CA (completer abort) -> SLVERR</p> <p>bit 3: Reserved</p> <p>bit 4: Reserved</p> <p>bit 5: 0: Completion Timeout -> DECERR 1: Completion Timeout -> SLVERR</p> |
| 9:5 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 4:3 | RW | 0x0 | <p>AMBA_ERROR_RESPONSE_CRS CRS Slave Error Response Mapping. Determines the AXI slave response for CRS completions.</p> <p>00: OKAY 01: OKAY with all FFFF_FFFF data for all CRS completions 10: OKAY with FFFF_0001 data for CRS completions to vendor ID read requests, OKAY with FFFF_FFFF data for all other CRS completions 11: SLVERR/DECERR (the AXI_ERROR_RESPONSE_MAP field determines the PCIe-to-AXI Slave error response mapping)</p> <p>Note: This register field is sticky.</p> |
| 2 | RW | 0x0 | <p>AMBA_ERROR_RESPONSE_VENDORID Vendor ID Non-existent Slave Error Response Mapping. Determines the AXI slave response for errors on reads to non-existent Vendor ID register.</p> <p>0: OKAY (with FFFF data). 1: SLVERR/DECERR (the AXI_ERROR_RESPONSE_MAP field determines the PCIe-to-AXI Slave error response mapping)</p> <p>Note: This register field is sticky.</p> |
| 1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | <p>AMBA_ERROR_RESPONSE_GLOBAL Global Slave Error Response Mapping. Determines the AXI slave response for all error scenarios on non-posted requests.</p> <p>0: OKAY (with FFFF data for non-posted requests) 1: SLVERR/DECERR (the AXI_ERROR_RESPONSE_MAP field determines the PCIe-to-AXI Slave error response mapping)</p> <p>The error response mapping is not applicable to Non-existent Vendor ID register reads.</p> <p>Note: This register field is sticky.</p> |

PL AMBA LINK TIMEOUT OFF

Address: Operational Base + offset (0x08d4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:9 | RO | 0x0 | reserved |
| 8 | RW | 0x0 | <p>LINK_TIMEOUT_ENABLE_DEFAULT Disable Flush. You can disable the flush feature by setting this field to "1".</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 7:0 | RW | 0x32 | <p>LINK_TIMEOUT_PERIOD_DEFAULT</p> <p>Timeout Value (ms). The timer will timeout and then flush the bridge TX request queues after this amount of time. The timer counts when there are pending outbound AXI slave interface requests and the PCIe TX link is not transmitting any of these requests. The timer is clocked by core_clk. For an M-PCIe configuration:</p> <p>Time unit of this field is 4 ms.</p> <p>Margin of error for RateA clock is < 1%.</p> <p>Margin of error for RateB clock is between 16% and 17%.</p> <p>Note: This register field is sticky.</p> |

PL AMBA ORDERING CTRL OFF

Address: Operational Base + offset (0x08d8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | <p>AX_MSTR_ZEROLREAD_FW</p> <p>AXI Master Zero Length Read Forward to the application.</p> <p>The DW PCIe controller AXI bridge is able to terminate in order with the Posted transactions the zero length read, implementing the PCIe express flush semantics of the Posted transactions.</p> <p>0x0: The zero length Read is terminated at the DW PCIe AXI bridge master</p> <p>0x1: The zero length Read is forward to the application.</p> |
| 6:5 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 4:3 | RW | 0x0 | <p>AX_MSTR_ORDR_P_EVENT_SEL AXI Master Posted Ordering Event Selector.</p> <p>This field selects how the master interface determines when a P write is completed when enforcing the PCIe ordering rule, "NP must not pass P" at the AXI Master Interface. The AXI protocol does not support ordering between channels. Therefore, NP reads can pass P on your AXI bus fabric. This can result in an ordering violation when the read overtakes a P that is going to the same address. Therefore, the bridge master does not issue any NP requests until all outstanding P writes reach their destination. It does this by waiting for the all of the write responses on the B channel. This can affect the performance of the master read channel. For scenarios where the interconnect serializes the AXI master "AW", "W" and "AR" channels, you can increase the performance by reducing the need to wait until the complete Posted transaction has effectively reached the application slave.</p> <p>00: B'last event: wait for the all of the write responses on the B channel thereby ensuring that the complete Posted transaction has effectively reached the application slave (default).</p> <p>01: AW'last event: wait until the complete Posted transaction has left the AXI address channel at the bridge master.</p> <p>10: W'last event: wait until the complete Posted transaction has left the AXI data channel at the bridge master.</p> <p>11: Reserved</p> |
| 2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | <p>AX_SNP_EN AXI Serialize Non-Posted Requests Enable. This field enables the AXI Bridge to serialize same ID Non-Posted Read/Write Requests on the wire. Serialization implies one outstanding same ID NP Read or Write on the wire and used to avoid AXI RAR and WAW hazards at the remote link partner.</p> |
| 0 | RO | 0x0 | reserved |

PL COHERENCY CONTROL 1 OFF

Address: Operational Base + offset (0x08e0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:2 | RW | 0x00000000 | <p>CFG_MEMTYPE_BOUNDARY_LOW_ADDR Boundary Lower Address For Memory Type. Bits [31:0] of dword-aligned address of the boundary for Memory type. The two lower address LSBs are "00". Addresses up to but not including this value are in the lower address space region; addresses equal or greater than this value are in the upper address space region.</p> <p>Note: This register field is sticky.</p> |
| 1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | RW | 0x0 | <p>CFG_MEMTYPE_VALUE Sets the memory type for the lower and upper parts of the address space: 0: lower = Peripheral; upper = Memory 1: lower = Memory type; upper = Peripheral Note: This register field is sticky.</p> |

PL COHERENCY CONTROL 2 OFF

Address: Operational Base + offset (0x08e4)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>CFG_MEMTYPE_BOUNDARY_HIGH_ADDR Boundary Upper Address For Memory Type. Bits [63:32] of the 64-bit dword-aligned address of the boundary for Memory type. Note: This register field is sticky.</p> |

PL COHERENCY CONTROL 3 OFF

Address: Operational Base + offset (0x08e8)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31 | RO | 0x0 | reserved |
| 30:27 | RW | 0x0 | <p>CFG_MSTR_AWCACHE_VALUE Master Write CACHE Signal Value. Value of the individual bits in mstr_awcache when CFG_MSTR_AWCACHE_MODE is '1'. Note: not applicable to message requests; for message requests the value of mstr_awcache is always "0000" Note: This register field is sticky.</p> |
| 26:23 | RO | 0x0 | reserved |
| 22:19 | RW | 0x0 | <p>CFG_MSTR_ARCACHE_VALUE Master Read CACHE Signal Value. Value of the individual bits in mstr_arcache when CFG_MSTR_ARCACHE_MODE is '1'. Note: This register field is sticky.</p> |
| 18:15 | RO | 0x0 | reserved |
| 14:11 | RW | 0x0 | <p>CFG_MSTR_AWCACHE_MODE Master Write CACHE Signal Behavior. Defines how the individual bits in mstr_awcache are controlled: 0: set automatically by the AXI master 1: set by the value of the corresponding bit of the CFG_MSTR_AWCACHE_VALUE field Note: for message requests the value of mstr_awcache is always "0000" regardless of the value of this bit Note: This register field is sticky.</p> |
| 10:7 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 6:3 | RW | 0x0 | CFG_MSTR_ARCACHE_MODE Master Read CACHE Signal Behavior. Defines how the individual bits in mstr_arcache are controlled: 0: set automatically by the AXI master 1: set by the value of the corresponding bit of the CFG_MSTR_ARCACHE_VALUE field Note: This register field is sticky. |
| 2:0 | RO | 0x0 | reserved |

PL_AXI_MSTR_MSG_ADDR_LOW_OFF

Address: Operational Base + offset (0x08f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RW | 0x00000 | CFG_AXIMSTR_MSG_ADDR_LOW Lower 20 bits of the programmable AXI address for Messages. Note: This register field is sticky. |
| 11:0 | RO | 0x000 | CFG_AXIMSTR_MSG_ADDR_LOW_RESERVED Reserved for future use. Note: This register field is sticky. |

PL_AXI_MSTR_MSG_ADDR_HIGH_OFF

Address: Operational Base + offset (0x08f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | CFG_AXIMSTR_MSG_ADDR_HIGH Upper 32 bits of the programmable AXI address for Messages. Note: This register field is sticky. |

PL_PCIE_VERSION_NUMBER_OFF

Address: Operational Base + offset (0x08f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-----------------------------------|
| 31:0 | RO | 0x3531302a | VERSION_NUMBER Version Number. |

PL_PCIE_VERSION_TYPE_OFF

Address: Operational Base + offset (0x08fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------|
| 31:0 | RO | 0x67612a2a | VERSION_TYPE Version Type. |

PL_MSIX_ADDRESS_MATCH_LOW_OFF

Address: Operational Base + offset (0x0940)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RW | 0x00000000 | MSIX_ADDRESS_MATCH_LOW MSI-X Address Match Low Address. Note: This register field is sticky. |
| 1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | MSIX_ADDRESS_MATCH_EN MSI-X Match Enable. Enable the MSI-X Address Match feature when the AXI bridge is present. Note: This register field is sticky. |

PL_MSIX_ADDRESS_MATCH_HIGH_OF

Address: Operational Base + offset (0x0944)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | MSIX_ADDRESS_MATCH_HIGH MSI-X Address Match High Address. Note: This register field is sticky. |

PL_MSIX_DOORBELL_OF

Address: Operational Base + offset (0x0948)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:29 | RO | 0x0 | reserved |
| 28:24 | WO | 0x00 | MSIX_DOORBELL_PF MSIX Doorbell Physical Function. This register determines the Physical Function for the MSI-X transaction. |
| 23:16 | WO | 0x00 | MSIX_DOORBELL_VF MSIX Doorbell Virtual Function. This register determines the Virtual Function for the MSI-X transaction. |
| 15 | WO | 0x0 | MSIX_DOORBELL_VF_ACTIVE MSIX Doorbell Virtual Function Active. This register determines whether a Virtual Function is used to generate the MSI-X transaction. |
| 14:12 | WO | 0x0 | MSIX_DOORBELL_TC MSIX Doorbell Traffic Class. This register determines which traffic class to generate the MSI-X transaction with. |
| 11 | RO | 0x0 | reserved |
| 10:0 | WO | 0x000 | MSIX_DOORBELL_VECTOR MSI-X Doorbell Vector. This register determines which vector to generate the MSI-X transaction for. |

PL_MSIX_RAM_CTRL_OF

Address: Operational Base + offset (0x094c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:26 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 25 | RW | 0x0 | <p>MSIX_RAM_CTRL_DBG_PBA MSIX PBA RAM Debug Mode. Use this bit to activate the debug mode and allow direct read/write access to the PBA. Use can also use the dbg_pba input to activate debug mode. Debug mode turns off the PF/VF/Offset-based addressing into the RAM and maps the entire table linearly from the base address of the BAR (indicated by the BIR) in function 0.</p> <p>Note: This register field is sticky.</p> |
| 24 | RW | 0x0 | <p>MSIX_RAM_CTRL_DBG_TABLE MSIX Table RAM Debug Mode. Use this bit to activate the debug mode and allow direct read/write access to the Table. Use can also use the dbg_table input to activate debug mode. Debug mode turns off the PF/VF/Offset-based addressing into the RAM and maps the entire table linearly from the base address of the BAR (indicated by the BIR) in function 0.</p> <p>Note: This register field is sticky.</p> |
| 23:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | <p>MSIX_RAM_CTRL_BYPASS MSIX RAM Control Bypass. The bypass field, when set, disables the internal generation of low power signals for both RAMs. It is up to the application to ensure the RAMs are in the proper power state before trying to access them. Moreover, the application needs to observe all timing requirements of the RAM low power signals before trying to use the MSIX functionality.</p> <p>Note: This register field is sticky.</p> |
| 15:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | <p>MSIX_RAM_CTRL_PBA_SD MSIX PBA RAM Shut Down. Set this bit to drive the cfg_msix_pba_sd output to signal your external logic to place the MSIX PBA RAM in Shut Down low-power mode.</p> <p>Note: This register field is sticky.</p> |
| 8 | RW | 0x0 | <p>MSIX_RAM_CTRL_PBA_DS MSIX PBA RAM Deep Sleep. Set this bit to drive the cfg_msix_pba_ds output to signal your external logic to place the MSIX PBA RAM in Deep Sleep low-power mode.</p> <p>Note: This register field is sticky.</p> |
| 7:2 | RO | 0x0 | reserved |
| 1 | RW | 0x0 | <p>MSIX_RAM_CTRL_TABLE_SD MSIX Table RAM Shut Down. Set this bit to drive the cfg_msix_table_sd output to signal your external logic to place the MSIX Table RAM in Shut Down low-power mode.</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 0 | RW | 0x0 | MSIX_RAM_CTRL_TABLE_DS MSIX Table RAM Deep Sleep. Set this bit to drive the cfg_msix_table_ds output to signal your external logic to place the MSIX Table RAM in Deep Sleep low-power mode. Note: This register field is sticky. |

PL_LTR_LATENCY_OFF

Address: Operational Base + offset (0x0b30)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31 | RW | 0x0 | NO_SNOOP_LATENCY_REQUIRE No Snoop Latency Requirement. Note: The access attributes of this field are as follows: Wire: R Dbi: R/W |
| 30:29 | RO | 0x0 | reserved |
| 28:26 | RW | 0x0 | NO_SNOOP_LATENCY_SCALE No Snoop Latency Scale. Note: The access attributes of this field are as follows: Wire: R Dbi: R/W |
| 25:16 | RW | 0x000 | NO_SNOOP_LATENCY_VALUE No Snoop Latency Value. Note: The access attributes of this field are as follows: Wire: R Dbi: R/W |
| 15 | RW | 0x0 | SNOOP_LATENCY_REQUIRE Snoop Latency Requirement. Note: The access attributes of this field are as follows: Wire: R Dbi: R/W |
| 14:13 | RO | 0x0 | reserved |
| 12:10 | RW | 0x0 | SNOOP_LATENCY_SCALE Snoop Latency Scale. Note: The access attributes of this field are as follows: Wire: R Dbi: R/W |
| 9:0 | RW | 0x000 | SNOOP_LATENCY_VALUE Snoop Latency Value. Note: The access attributes of this field are as follows: Wire: R Dbi: R/W |

PL_AUX_CLK_FREQ_OFF

Address: Operational Base + offset (0x0b40)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:10 | RO | 0x0 | reserved |
| 9:0 | RW | 0x018 | AUX_CLK_FREQ The aux_clk frequency in MHz. This value is used to provide a 1 us reference for counting time during low-power states with aux_clk when the PHY has removed the pipe_clk. Frequencies lower than 1 MHz are possible but with a loss of accuracy in the time counted. If the actual frequency (f) of aux_clk does not exactly match the programmed frequency (f_prog), then there is an error in the time counted by the controller that can be expressed in percentage as: err% = (f_prog/f-1)*100. For example if f=2.5 MHz and f_prog=3 MHz, then err% =(3/2.5-1)*100 =20%, meaning that the time counted by the controller on aux_clk will be 20% greater than the time in us programmed in the corresponding time register (for example T_POWER_ON). Note: This register field is sticky. |

PL L1 SUBSTATES OFF

Address: Operational Base + offset (0x0b44)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | reserved |
| 7:6 | RW | 0x3 | L1SUB_T_PCLKACK Max delay (in 1us units) between a MAC request to remove the clock on mac_phy_pclkreq_n and a PHY response on phy_mac_pclkack_n. If the PHY does not respond within this time the request is aborted. Range is 0..3 Note: This register field is sticky. |
| 5:2 | RW | 0x4 | L1SUB_T_L1_2 Duration (in 1us units) of L1.2. Range is 0.15. Note: The timeout value can vary by 50%. Note: This register field is sticky. |
| 1:0 | RW | 0x2 | L1SUB_T_POWER_OFF Duration (in 1us units) of L1.2.Entry. Range is 0.3. Note: The timeout value can vary by 50%. Note: This register field is sticky. |

PL PIPE RELATED OFF

Address: Operational Base + offset (0x0b90)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:9 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 8 | RW | 0x0 | <p>PIPE_GARBAGE_DATA_MODE PIPE Garbage Data Mode.</p> <p>0: PIPE Spec compliant mode: The MAC discards any symbols received after the electrical idle ordered-set until RxValid is deasserted.</p> <p>1: Special PHY Support mode: The MAC discards any symbols received after the electrical idle ordered-set until when any of the following three conditions are true:</p> <ul style="list-style-type: none"> RxValid is deasserted a valid RxStartBlock is received at 128b/130b encoding a valid COM symbol is received at 8b/10b encoding <p>Note: This register field is sticky.</p> |
| 7:0 | RO | 0x0 | reserved |

10.5.32 Shadow Block Registers Summary

| Name | Offset | Size | Reset Value | Description |
|---|----------|------|-------------|--|
| SB BAR0 MASK REG | 0x100010 | W | 0xffffffff | BAR0 Mask Register. |
| SB BAR1 MASK REG | 0x100014 | W | 0xffffffff | BAR1 Mask Register. |
| SB BAR2 MASK REG | 0x100018 | W | 0xffffffff | BAR2 Mask Register. |
| SB BAR3 MASK REG | 0x10001c | W | 0xffffffff | BAR3 Mask Register. |
| SB BAR4 MASK REG | 0x100020 | W | 0xffffffff | BAR4 Mask Register. |
| SB BAR5 MASK REG | 0x100024 | W | 0xffffffff | BAR5 Mask Register. |
| SB EXP ROM BAR MASK REG | 0x100030 | W | 0x0001ffff | Expansion ROM BAR Mask Register. |
| SB SHADOW PCI MSIX CAP ID NEXT CTRL REG | 0x1000b0 | W | 0x003f0000 | MSI-X Capability ID, Next Pointer, Control shadow Registers. |

Notes: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.33 Shadow Block Detail Register Description

SB BAR0 MASK REG

Address: Operational Base + offset (0x100010)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:1 | WO | 0x7fffffff | <p>PCI_TYPE0_BAR0_MASK BAR0 Mask.</p> <p>Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky.</p> |
| 0 | WO | 0x1 | <p>PCI_TYPE0_BAR0_ENABLED BAR0 Mask Enabled.</p> <p>Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky.</p> |

SB_BAR1_MASK_REG

Address: Operational Base + offset (0x100014)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:1 | WO | 0x7fffffff | <p>PCI_TYPE0_BAR1_MASK BAR1 Mask. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky.</p> |
| 0 | WO | 0x1 | <p>PCI_TYPE0_BAR1_ENABLED BAR1 Mask Enabled. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky.</p> |

SB_BAR2_MASK_REG

Address: Operational Base + offset (0x100018)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:1 | WO | 0x7fffffff | <p>PCI_TYPE0_BAR2_MASK BAR2 Mask. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky.</p> |
| 0 | WO | 0x1 | <p>PCI_TYPE0_BAR2_ENABLED BAR2 Mask Enabled. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky.</p> |

SB_BAR3_MASK_REG

Address: Operational Base + offset (0x10001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | WO | 0x7fffffff | PCI_TYPE0_BAR3_MASK BAR3 Mask. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky. |
| 0 | WO | 0x1 | PCI_TYPE0_BAR3_ENABLED BAR3 Mask Enabled. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky. |

SB BAR4 MASK REG

Address: Operational Base + offset (0x100020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | WO | 0x7fffffff | PCI_TYPE0_BAR4_MASK BAR4 Mask. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky. |
| 0 | WO | 0x1 | PCI_TYPE0_BAR4_ENABLED BAR4 Mask Enabled. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky. |

SB BAR5 MASK REG

Address: Operational Base + offset (0x100024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | WO | 0x7fffffff | PCI_TYPE0_BAR5_MASK BAR5 Mask. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky. |
| 0 | WO | 0x1 | PCI_TYPE0_BAR5_ENABLED BAR5 Mask Enabled. Note: The access attributes of this field are as follows: Wire: No access Dbi: W (sticky) Note: This register field is sticky. |

SB EXP ROM BAR MASK REG

Address: Operational Base + offset (0x100030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | WO | 0x0000ffff | PCI_TYPE0_BAR0_MASK Expansion ROM Mask. Note: The access attributes of this field are as follows: Wire: No access Dbi: if ROM_BAR_ENABLED && ROM_MASK_WRITABLE then W Note: This register field is sticky. |
| 0 | WO | 0x1 | ROM_BAR_ENABLED Expansion ROM Bar Mask Register Enabled. Note: The access attributes of this field are as follows: Wire: No access Dbi: if ROM_MASK_WRITABLE then W Note: This register field is sticky. |

SB SHADOW PCI MSIX CAP ID NEXT CTRL REG

Address: Operational Base + offset (0x1000b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:27 | RO | 0x0 | reserved |
| 26:16 | RW | 0x03f | PCI_MSIX_TABLE_SIZE MSI-X Table Size in the shadow register. Note: The access attributes of this field are as follows: Wire: No access Dbi: if (DBI_RO_WR_EN == 1) then R/W else R Note: This register field is sticky. |
| 15:0 | RO | 0x0 | reserved |

10.5.34 ATU Registers Summary

ATU Port Logic Structure Registers Follow the link for the register to see a detailed description of the register.

| Name | Offset | Size | Reset Value | Description |
|--|------------------------------|-------------|--------------------|---|
| IATU REGION CTRL 1_O FF_OUTBOUND_i | 0x300000 +i*0x200 +0x0 | W | 0x00000000 | iATU Region Control 1 Register. (for i = 0; i <= 15) |
| IATU REGION CTRL 2_O FF_OUTBOUND_i | 0x300000 +i*0x200 +0x4 | W | 0x00000000 | iATU Region Control 2 Register. (for i = 0; i <= 15) |
| IATU_LWR_BASE_ADDR OFF_OUTBOUND_i | 0x300000 +i*0x200 +0x8 | W | 0x00000000 | iATU Lower Base Address Register. (for i = 0; i <= 15) |
| IATU_UPPER_BASE_ADDR OFF_OUTBOUND_i | 0x300000 +i*0x200 +0xc | W | 0x00000000 | iATU Upper Base Address Register. (for i = 0; i <= 15) |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------------|--------------------------|------|-------------|--|
| IATU LIMIT ADDR OFF OUTBOUND i | 0x300000 +i*0x200 +0x10 | W | 0x0000ffff | iATU Limit Address Register. (for i = 0; i <= 15) |
| IATU LWR TARGET ADDR OFF OUTBOUND i | 0x300000 +i*0x200 +0x14 | W | 0x00000000 | iATU Lower Target Address Register. (for i = 0; i <= 15) |
| IATU UPPER TARGET ADDR OFF OUTBOUND i | 0x300000 +i*0x200 +0x18 | W | 0x00000000 | iATU Upper Target Address Register. (for i = 0; i <= 15) |
| IATU REGION CTRL 1 OFF INBOUND i | 0x300000 +i*0x200 +0x100 | W | 0x00000000 | iATU Region Control 1 Register. (for i = 0; i <= 15) |
| IATU REGION CTRL 2 OFF INBOUND i | 0x300000 +i*0x200 +0x104 | W | 0x00000000 | iATU Region Control 2 Register. (for i = 0; i <= 15) |
| IATU LWR BASE ADDR OFF INBOUND i | 0x300000 +i*0x200 +0x108 | W | 0x00000000 | iATU Lower Base Address Register. (for i = 0; i <= 15) |
| IATU UPPER BASE ADDR OFF INBOUND i | 0x300000 +i*0x200 +0x10c | W | 0x00000000 | iATU Upper Base Address Register. (for i = 0; i <= 15) |
| IATU LIMIT ADDR OFF INBOUND i | 0x300000 +i*0x200 +0x110 | W | 0x0000ffff | iATU Limit Address Register. (for i = 0; i <= 15) |
| IATU LWR TARGET ADDR OFF INBOUND i | 0x300000 +i*0x200 +0x114 | W | 0x00000000 | iATU Lower Target Address Register. (for i = 0; i <= 15) |

Notes: **S**-ize: **B**- Byte (8 bits) access, **H****W**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

10.5.35 ATU Detail Register Description

IATU REGION CTRL 1 OFF OUTBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x0)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:23 | RO | 0x0 | reserved |
| 22:20 | RW | 0x0 | CTRL_1_FUNC_NUM Function Number. Note: This register field is sticky. |
| 19:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | INCREASE_REGION_SIZE Increase the maximum ATU Region size. When set, the maximum ATU Region size is determined by CX_ATU_MAX_REGION_SIZE. When clear, the maximum ATU Region size is 4 GB (default). Note: This register field is sticky. |
| 12:11 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 10:9 | RW | 0x0 | ATTR When the address of an outbound TLP is matched to this region, then the ATTR field of the TLP is changed to the value in this register. Note: This register field is sticky. |
| 8 | RW | 0x0 | TD This is a reserved field. Do not use. |
| 7:5 | RW | 0x0 | TC When the address of an outbound TLP is matched to this region, then the TC field of the TLP is changed to the value in this register. Note: This register field is sticky. |
| 4:0 | RW | 0x00 | TYPE When the address of an outbound TLP is matched to this region, then the TYPE field of the TLP is changed to the value in this register. Note: This register field is sticky. |

IATU REGION CTRL 2 OFF OUTBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | REGION_EN Region Enable. This bit must be set to '1' for address translation to take place. Note: This register field is sticky. |
| 30 | RO | 0x0 | reserved |
| 29 | RW | 0x0 | INVERT_MODE Invert Mode. When set the address matching region is inverted. Therefore, an address match occurs when the untranslated address is in the region outside the defined range (Base Address to Limit Address). Note: This register field is sticky. |
| 28 | RW | 0x0 | CFG_SHIFT_MODE CFG Shift Mode. The iATU uses bits [27:12] of the untranslated address (on the XALI0/1/2 interface or AXI slave interface address) to form the BDF number of the outgoing CFG TLP. This supports the Enhanced Configuration Address Mapping (ECAM) mechanism (Section 7.2.2 of the PCI Express Base 3.1 Specification, revision 1.0) by allowing all outgoing I/O and MEM TLPs (that have been translated to CFG) to be mapped from memory space into any 256 MB region of the PCIe configuration space. Note: This register field is sticky. |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 27 | RW | 0x0 | DMA_BYPASS DMA Bypass Mode. Allows request TLPs which are initiated by the DMA controller to pass through the iATU untranslated. Note: This register field is sticky. |
| 26:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | HEADER_SUBSTITUTE_EN Header Substitute Enable. When enabled and region address is matched, the iATU fully substitutes bytes 8-11 (for 3 DWORD header) or bytes 12-15 (for 4 DWORD header) of the outbound TLP header with the contents of the LWR_TARGET_RW field in iATU_LWR_TARGET_ADDR_OFF_OUTBOUND_i. 1: LWR_TARGET_RW in the iATU_LWR_TARGET_ADDR_OFF_OUTBOUND_i register is used to fill bytes 8-to-11 (for 3 DWORD header) or bytes 12-to-15 (for 4 DWORD header) of the translated TLP header. 0: LWR_TARGET_RW in the iATU_LWR_TARGET_ADDR_OFF_OUTBOUND_i register forms the new address of the translated region. Note: This register field is sticky. |
| 22 | RW | 0x0 | INHIBIT_PAYLOAD Inhibit TLP Payload Data for TLP's in Matched Region; assign iATU region to be TLP without data. When enabled and region address is matched, the iATU marks all TLPs as having no payload data by forcing the TLP header Fmt[1] bit =0, regardless of the application inputs such as slv_wstrb. 1: Fmt[1] =0 so that only TLP type without data is sent. For example, a Msg instead of MsgD will be sent. 0: Fmt[1] =0/1 so that TLPs with or without data can be sent. Note: This register field is sticky. |
| 21 | RO | 0x0 | reserved |
| 20 | RW | 0x0 | SNP Serialize Non-Posted Requests. In this mode, when the AXI Bridge is populated, same AXI ID Non-Posted Read/Write Requests are transmitted on the wire if there are no other same ID Non-Posted Requests outstanding. Note: This register field is sticky. |
| 19 | RW | 0x0 | FUNC_BYPASS Function Number Translation Bypass. In this mode, the function number of the translated TLP is taken from your application transmit interface and not from the CTRL_1_FUNC_NUM field of the "iATU Control 1 Register" or the VF_NUMBER field of the "iATU Control 3 Register." Note: This register field is sticky. |
| 18:17 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 16 | RW | 0x0 | <p>TAG_SUBSTITUTE_EN TAG Substitute Enable. When enabled and region address is matched, the iATU substitutes the TAG field of the outbound TLP header with the contents of the TAG field in this register. The expected usage scenario is translation from AXI MWr to Vendor Defined Msg/MsgD. Note: This register field is sticky.</p> |
| 15:8 | RW | 0x00 | <p>TAG TAG. The substituted TAG field (byte 6) in the outgoing TLP header when TAG_SUBSTITUTE_EN is set. Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>MSG_CODE MSG TLPs (Message Code). When the address of an outbound TLP is matched to this region, and the translated TLP TYPE field is Msg or MsgD; then the message field of the TLP is changed to the value in this register. Note: This register field is sticky.</p> |

IATU LWR BASE ADDR OFF OUTBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x8)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | <p>LWR_BASE_RW Forms bits [31:n] of the start address of the address region to be translated. n is log2(CX_ATU_MIN_REGION_SIZE) Note: This register field is sticky.</p> |
| 15:0 | RO | 0x0000 | <p>LWR_BASE_HW Forms bits [n-1:0] of the start address of the address region to be translated. The start address must be aligned to a CX_ATU_MIN_REGION_SIZE kB boundary, so these bits are always 0. A write to this location is ignored by the PCIe controller. n is log2(CX_ATU_MIN_REGION_SIZE)</p> |

IATU UPPER BASE ADDR OFF OUTBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0xC)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>UPPER_BASE_RW Forms bits [63:32] of the start (and end) address of the address region to be translated. In systems with a 32-bit address space, this register is not used and therefore writing to this register has no effect.</p> <p>Note: This register field is sticky.</p> |

IATU LIMIT ADDR OFF OUTBOUND_i

Address: Operational Base + offset (0x300000+i*0x20+0x10)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0000 | <p>LIMIT_ADDR_RW Forms upper bits of the end address of the address region to be translated.</p> <p>Note: This register field is sticky.</p> |
| 15:0 | RW | 0xffff | <p>LIMIT_ADDR_HW Forms lower bits of the end address of the address region to be translated. The end address must be aligned to a CX_ATU_MIN_REGION_SIZE kB boundary, so these bits are always all ones. A write to this location is ignored by the PCIe controller.</p> |

IATU LWR TARGET ADDR OFF OUTBOUND_i

Address: Operational Base + offset (0x300000+i*0x20+0x14)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | <p>LWR_TARGET_RW_OUTBOUND When HEADER_SUBSTITUTE_EN in IATU_REGION_CTRL_2_OFF_OUTBOUND_i is '0' (normal operation): LWR_TARGET_RW[31:n] forms MSB's of the Lower Target part of the new address of the translated region; LWR_TARGET_RW[n-1:0] are not used. (The start address must be aligned to a CX_ATU_MIN_REGION_SIZE kB boundary, so the lower bits of the start address of the new address of the translated region (bits n-1:0) are always '0'). n is log2(CX_ATU_MIN_REGION_SIZE). When HEADER_SUBSTITUTE_EN in IATU_REGION_CTRL_2_OFF_OUTBOUND_i is '1': LWR_TARGET_RW[31:0] forms bytes 8-11 (for 3 dword header) or bytes 12-15 (for 4 dword header) of the outbound TLP header. Usage scenarios include the transmission of Vendor Defined Messages where the controller determines the content of bytes 12 to 15 of the TLP header.</p> <p>Note: This register field is sticky.</p> |

IATU UPPER TARGET ADDR OFF OUTBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x18)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | UPPER_TARGET_RW Forms bits [63:32] of the start address (Upper Target part) of the new address of the translated region. Note: This register field is sticky. |

IATU REGION CTRL 1 OFF INBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:23 | RO | 0x0 | reserved |
| 22:20 | RW | 0x0 | CTRL_1_FUNC_NUM Function Number. Note: This register field is sticky. |
| 19:14 | RO | 0x0 | reserved |
| 13 | RW | 0x0 | INCREASE_REGION_SIZE Increase the maximum ATU Region size. When set, the maximum ATU Region size is determined by CX_ATU_MAX_REGION_SIZE. When clear, the maximum ATU Region size is 4 GB (default). Note: This register field is sticky. |
| 12:11 | RO | 0x0 | reserved |
| 10:9 | RW | 0x0 | ATTR When the ATTR field of an inbound TLP is matched to this value, then address translation proceeds (when all other enabled field-matches are successful). This check is only performed if the "ATTR Match Enable" bit of the "iATU Control 2 Register" is set. Note: This register field is sticky. |
| 8 | RW | 0x0 | TD When the TD field of an inbound TLP is matched to this value, then address translation proceeds (when all other enabled field-matches are successful). This check is only performed if the "TD Match Enable" bit of the "iATU Control 2 Register" is set. Note: This register field is sticky. |
| 7:5 | RW | 0x0 | TC When the TC field of an inbound TLP is matched to this value, then address translation proceeds (when all other enabled field-matches are successful). This check is only performed if the "TC Match Enable" bit of the "iATU Control 2 Register" is set. Note: This register field is sticky. |
| 4:0 | RW | 0x00 | TYPE When the TYPE field of an inbound TLP is matched to this value, then address translation proceeds (when all other enabled field-matches are successful). Note: This register field is sticky. |

IATU REGION CTRL 2 OFF INBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x104)

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 31 | RW | 0x0 | <p>REGION_EN</p> <p>Region Enable. This bit must be set to '1' for address translation to take place.</p> <p>Note: This register field is sticky.</p> |
| 30 | RW | 0x0 | <p>MATCH_MODE</p> <p>Match Mode. Determines Inbound matching mode for TLPs. The mode depends on the type of TLP that is received as follows:</p> <p>For MEM-I/O TLPs, this field is interpreted as follows:</p> <p>0: Address Match Mode. The iATU operates using addresses as in the outbound direction. The Region Base and Limit Registers must be setup.</p> <p>1: BAR Match Mode. BAR matching is used. The "BAR Number" field is relevant. Not used for RC.</p> <p>For CFG0 TLPs, this field is interpreted as follows:</p> <p>0: Routing ID Match Mode. The iATU interprets the Routing ID (Bytes 8 to 11 of TLP header) as an address. This corresponds to the upper 16 bits of the address in MEM-I/O transactions. The Routing ID of the TLP must be within the base and limit of the iATU region for matching to proceed.</p> <p>1: Accept Mode. The iATU accepts all CFG0 transactions as address matches. The routing ID in the CFG0 TLP is ignored. This is useful as all received CFG0 TLPs should be processed regardless of the Bus number.</p> <p>For MSG/MSGD TLPs, this field is interpreted as follows:</p> <p>0: Address Match Mode. The iATU treats the third dword and fourth dword of the inbound MSG/MSGD TLP as an address and it is matched against the Region Base and Limit Registers.</p> <p>1: Vendor ID Match Mode. This mode is relevant for ID-routed Vendor Defined Messages. The iATU ignores the Routing ID (Bus, Device, Function) in bits [31:16] of the third dword of the TLP header, but matches against the Vendor ID in bits [15:0] of the third dword of the TLP header. Bits [15:0] of the Region Upper Base register should be programmed with the required Vendor ID. The lower Base and Limit Register should be programmed to translate TLPs based on vendor specific information in the fourth dword of the TLP header.</p> |
| 29 | RW | 0x0 | <p>INVERT_MODE</p> <p>Invert Mode. When set the address matching region is inverted. Therefore, an address match occurs when the untranslated address is in the region outside the defined range (Base Address to Limit Address).</p> <p>Note: This register field is sticky.</p> |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 28 | RW | 0x0 | <p>CFG_SHIFT_MODE CFG Shift Mode. This is useful for CFG transactions where the PCIe configuration mechanism maps bits [27:12] of the address to the bus/device and function number. This allows a CFG configuration space to be located in any 256MB window of your application memory space using a 28-bit effective address. Shifts bits [31:16] of the untranslated address to form bits [27:12] of the translated address.</p> <p>Note: This register field is sticky.</p> |
| 27 | RW | 0x0 | <p>FUZZY_TYPE_MATCH_CODE Fuzzy Type Match Mode. When enabled, the iATU relaxes the matching of the TLP TYPE field against the expected TYPE field so that CfgRd0 and CfgRd1 TLPs are seen as identical. Similarly with CfgWr0 and CfgWr1. MWr, MRd and MRdLk TLPs are seen as identical The Routing field of Msg/MsgD TLPs is ignored FetchAdd, Swap and CAS are seen as identical. For example, CFG0 in the TYPE field in the "iATU Control 1 Register" matches against an inbound CfgRd0, CfgRd1, CfgWr0 or CfgWr1 TLP.</p> <p>Note: This register field is sticky.</p> |
| 26 | RO | 0x0 | reserved |
| 25:24 | RW | 0x0 | <p>RESPONSE_CODE Response Code. Defines the type of response to give for accesses matching this region. This overrides the normal RADM filter response. Note that this feature is not available for any region where Single Address Location Translate is enabled.</p> <p>00 - Normal RADM filter response is used. 01 - Unsupported request (UR) 10 - Completer abort (CA) 11 - Not used / undefined / reserved.</p> <p>Note: This register field is sticky.</p> |
| 23 | RW | 0x0 | <p>SINGLE_ADDR_LOC_TRANS_EN Single Address Location Translate Enable. When enabled, Rx TLPs can be translated to a single address location as determined by the target address register of the iATU region. The main usage scenario is translation of Messages (such as Vendor Defined or ATS Messages) to MWr TLPs when the AXI bridge is enabled.</p> <p>Note: This register field is sticky.</p> |
| 22 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 21 | RW | 0x0 | <p>MSG_CODE_MATCH_EN Message Code Match Enable (Msg TLPS). Ensures that a successful message Code TLP field comparison match (see Message Code field of the "iATU Control 2 Register") occurs (in MSG transactions) for address translation to proceed.</p> <p>ST Match Enable (Mem TLPS). Ensures that a successful ST TLP field comparison match (see ST field of the "iATU Control 2 Register") occurs (in MEM transactions) for address translation to proceed. Only Valid when the CX_TPH_ENABLE configuration parameter is 1</p> <p>Note: This register field is sticky.</p> |
| 20 | RO | 0x0 | reserved |
| 19 | RW | 0x0 | <p>FUNC_NUM_MATCH_EN Function Number Match Enable. Ensures that a successful Function Number TLP field comparison match (see Function Number field of the "iATU Control 1 Register") occurs (in MEM-I/O and CFG0/CFG1 transactions) for address translation to proceed.</p> <p>Note: This register field is sticky.</p> |
| 18:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | <p>ATTR_MATCH_EN ATTR Match Enable. Ensures that a successful ATTR TLP field comparison match (see ATTR field of the "iATU Control 1 Register") occurs for address translation to proceed.</p> <p>Note: This register field is sticky.</p> |
| 15 | RW | 0x0 | <p>TD_MATCH_EN TD Match Enable. Ensures that a successful TD TLP field comparison match (see TD field of the "iATU Control 1 Register") occurs for address translation to proceed.</p> <p>Note: This register field is sticky.</p> |
| 14 | RW | 0x0 | <p>TC_MATCH_EN TC Match Enable. Ensures that a successful TC TLP field comparison match (see TC field of the "iATU Control 1 Register") occurs for address translation to proceed.</p> <p>Note: This register field is sticky.</p> |
| 13 | RW | 0x0 | <p>MSG_TYPE_MATCH_MODE Message Type Match Mode. When enabled, and if single address location translate enable is set, then inbound TLPS of type MSG/MSGd which match the type field of the iatu_region_ctrl_1_OFF_inbound register ($=>$TYPE[4:3]=2'b10) will be translated. Message type match mode overrides any value of MATCH_MODE field in this register. Usage scenarios for this are translation of VDM or ATS messages when AXI bridge is configured on client interface.</p> <p>Note: This register field is sticky.</p> |
| 12:11 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 10:8 | RW | 0x0 | <p>BAR_NUM BAR Number. When the BAR number of an inbound MEM or IO TLP " that is matched by the normal internal BAR address matching mechanism " is the same as this field, address translation proceeds (when all other enabled field-matches are successful). This check is only performed if the "Match Mode" bit of the "iATU Control 2 Register" is set.</p> <p>000b - BAR0 001b - BAR1 010b - BAR2 011b - BAR3 100b - BAR4 101b - BAR5 110b - ROM 111b - reserved</p> <p>IO translation would require either 00100b or 00101b in the inbound TLP TYPE; the BAR Number set in the range 000b - 101b and that BAR configured as an IO BAR.</p> <p>Note: This register field is sticky.</p> |
| 7:0 | RW | 0x00 | <p>MSG_CODE MSG TLPs: (Message Code). When the TYPE field of an inbound Msg/MsgD TLP is matched to this value, then address translation proceeds (when all other enabled field-matches are successful). This check is only performed if the "Message Code Match Enable" bit of the "iATU Control 2 Register" is set.</p> <p>Note: This register field is sticky.</p> |

IATU_LWR_BASE_ADDR_OFF_INBOUND_i

Address: Operational Base + offset (0x300000+i*0x20+0x108)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | <p>LWR_BASE_RW Forms bits [31:n] of the start address of the address region to be translated. n is log2(CX_ATU_MIN_REGION_SIZE) Note: This register field is sticky.</p> |
| 15:0 | RO | 0x0000 | <p>LWR_BASE_HW Forms bits [n-1:0] of the start address of the address region to be translated. The start address must be aligned to a CX_ATU_MIN_REGION_SIZE kB boundary, so these bits are always 0. A write to this location is ignored by the PCIe controller. n is log2(CX_ATU_MIN_REGION_SIZE)</p> |

IATU UPPER BASE ADDR OFF INBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x10C)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | UPPER_BASE_RW Forms bits [63:32] of the start (and end) address of the address region to be translated. Note: This register field is sticky. |

IATU LIMIT ADDR OFF INBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | LIMIT_ADDR_RW Forms upper bits of the end address of the address region to be translated. Note: This register field is sticky. |
| 15:0 | RO | 0x0000 | LIMIT_ADDR_HW Forms lower bits of the end address of the address region to be translated. The end address must be aligned to a CX_ATU_MIN_REGION_SIZE kB boundary, so these bits are always all ones. A write to this location is ignored by the PCIe controller. |

IATU LWR TARGET ADDR OFF INBOUND i

Address: Operational Base + offset (0x300000+i*0x20+0x114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | LWR_TARGET_RW Forms MSB's of the Lower Target part of the new address of the translated region. These bits are always '0'. Field size depends on log2(CX_ATU_MIN_REGION_SIZE) in address match mode. Field size depends on log2(BAR_MASK+1) in BAR match mode. Note: This register field is sticky. |
| 15:0 | RO | 0x0000 | LWR_TARGET_HW Forms the LSB's of the Lower Target part of the new address of the translated region. The start address must be aligned to a CX_ATU_MIN_REGION_SIZE kB boundary (in address match mode); and to the Bar size boundary (in BAR match mode) so that these bits are always '0'. If the BAR is smaller than the iATU region size, then the iATU target address must align to the iATU region size; otherwise it must align to the BAR size. A write to this location is ignored by the PCIe controller. Field size depends on log2(CX_ATU_MIN_REGION_SIZE) in address match mode. Field size depends on log2(BAR_MASK+1) in BAR match mode. |

10.5.36 DMA Registers Summary

| Name | Offset | Size | Reset Value | Description |
|---|----------|------|-------------|--|
| DMA CTRL DATA ARB_P <u>RIOR OFF</u> | 0x380000 | W | 0x000000688 | DMA Arbitration Scheme for TRGT1 Interface. |
| DMA CTRL OFF | 0x380008 | W | 0x00010001 | DMA Number of Channels Register. |
| DMA WRITE ENGINE EN <u>OFF</u> | 0x38000c | W | 0x000000000 | DMA Write Engine Enable Register. |
| DMA WRITE DOORBELL <u>OFF</u> | 0x380010 | W | 0x000000000 | DMA Write Doorbell Register. |
| DMA WRITE CHANNEL A RB_WEIGHT_LOW OFF | 0x380018 | W | 0x00008421 | DMA Write Engine Channel Arbitration Weight Low Register. |
| DMA WRITE CHANNEL A RB_WEIGHT_HIGH OFF | 0x38001c | W | 0x00008421 | DMA Write Engine Channel Arbitration Weight High Register. |
| DMA READ ENGINE EN <u>OFF</u> | 0x38002c | W | 0x000000000 | DMA Read Engine Enable Register. |
| DMA READ DOORBELL_O FF | 0x380030 | W | 0x000000000 | DMA Read Doorbell Register. |
| DMA READ CHANNEL_AR B_WEIGHT_LOW OFF | 0x380038 | W | 0x00008421 | DMA Read Engine Channel Arbitration Weight Low Register. |
| DMA READ CHANNEL_AR B_WEIGHT_HIGH OFF | 0x38003c | W | 0x00008421 | DMA Read Engine Channel Arbitration Weight High Register. |
| DMA WRITE INT STATUS <u>OFF</u> | 0x38004c | W | 0x000000000 | DMA Write Interrupt Status Register. |
| DMA WRITE INT MASK <u>OFF</u> | 0x380054 | W | 0x00010001 | DMA Write Interrupt Mask Register. |
| DMA WRITE INT CLEAR <u>OFF</u> | 0x380058 | W | 0x000000000 | DMA Write Interrupt Clear Register. |
| DMA WRITE ERR STATU S OFF | 0x38005c | W | 0x000000000 | DMA Write Error Status Register |
| DMA WRITE DONE IMW R_LOW OFF | 0x380060 | W | 0x000000000 | DMA Write Done IMWr Address Low Register. |
| DMA WRITE DONE IMW R_HIGH OFF | 0x380064 | W | 0x000000000 | DMA Write Done IMWr Interrupt Address High Register. |
| DMA WRITE ABORT IMW R_LOW OFF | 0x380068 | W | 0x000000000 | DMA Write Abort IMWr Address Low Register. |
| DMA WRITE ABORT IMW R_HIGH OFF | 0x38006c | W | 0x000000000 | DMA Write Abort IMWr Address High Register. |
| DMA WRITE CH01_IMWR DATA OFF | 0x380070 | W | 0x000000000 | DMA Write Channel 1 and 0 IMWr Data Register. |
| DMA WRITE CH23_IMWR DATA OFF | 0x380074 | W | 0x000000000 | DMA Write Channel 3 and 2 IMWr Data Register. |
| DMA WRITE CH45_IMWR DATA OFF | 0x380078 | W | 0x000000000 | DMA Write Channel 5 and 4 IMWr Data Register. |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------------|----------|------|-------------|--|
| DMA WRITE CH67 IMWR DATA OFF | 0x38007c | W | 0x00000000 | DMA Write Channel 7 and 6 IMWr Data Register. |
| DMA WRITE LINKED LIST_ERR EN OFF | 0x380090 | W | 0x00000000 | DMA Write Linked List Error Enable Register. |
| DMA READ INT STATUS OFF | 0x3800a0 | W | 0x00000000 | DMA Read Interrupt Status Register. |
| DMA READ INT MASK OFF | 0x3800a8 | W | 0x00010001 | DMA Read Interrupt Mask Register. |
| DMA READ INT CLEAR OFF | 0x3800ac | W | 0x00000000 | DMA Read Interrupt Clear Register. |
| DMA READ ERR STATUS LOW OFF | 0x3800b4 | W | 0x00000000 | DMA Read Error Status Low Register. |
| DMA READ ERR STATUS HIGH OFF | 0x3800b8 | W | 0x00000000 | DMA Read Error Status High Register. |
| DMA READ LINKED LIST_ERR EN OFF | 0x3800c4 | W | 0x00000000 | DMA Read Linked List Error Enable Register. |
| DMA READ DONE IMWR LOW OFF | 0x3800cc | W | 0x00000000 | DMA Read Done IMWr Address Low Register. |
| DMA READ DONE IMWR HIGH OFF | 0x3800d0 | W | 0x00000000 | DMA Read Done IMWr Address High Register. |
| DMA READ ABORT IMWR LOW OFF | 0x3800d4 | W | 0x00000000 | DMA Read Abort IMWr Address Low Register. |
| DMA READ ABORT IMWR HIGH OFF | 0x3800d8 | W | 0x00000000 | DMA Read Abort IMWr Address High Register. |
| DMA READ CH01 IMWR DATA OFF | 0x3800dc | W | 0x00000000 | DMA Read Channel 1 and 0 IMWr Data Register. |
| DMA READ CH23 IMWR DATA OFF | 0x3800e0 | W | 0x00000000 | DMA Read Channel 3 and 2 IMWr Data Register. |
| DMA READ CH45 IMWR DATA OFF | 0x3800e4 | W | 0x00000000 | DMA Read Channel 5 and 4 IMWr Data Register. |
| DMA READ CH67 IMWR DATA OFF | 0x3800e8 | W | 0x00000000 | DMA Read Channel 7 and 6 IMWr Data Register. |
| DMA WRITE ENGINE HS HAKE_CNT LOW OFF | 0x380108 | W | 0x00000000 | DMA Write Engine Handshake Counter Channel 0/1/2/3 Register. |
| DMA WRITE ENGINE HS HAKE_CNT HIGH OFF | 0x38010c | W | 0x00000000 | DMA Write Engine Handshake Counter Channel 4/5/6/7 Register. |
| DMA READ ENGINE HSH AKE_CNT LOW OFF | 0x380118 | W | 0x00000000 | DMA Read Engine Handshake Counter Channel 0/1/2/3 Register. |
| DMA READ ENGINE HSH AKE_CNT HIGH OFF | 0x38011c | W | 0x00000000 | DMA Read Engine Handshake Counter Channel 4/5/6/7 Register. |

| Name | Offset | Size | Reset Value | Description |
|-------------------------------|----------|------|-------------|--|
| DMA CH CONTROL1 OFF WRCH_0 | 0x380200 | W | 0x00000000 | DMA Write Channel Control 1 Register. |
| DMA TRANSFER SIZE OF F_WRCH_0 | 0x380208 | W | 0x00000000 | DMA Write Transfer Size Register. |
| DMA SAR LOW OFF WR CH_0 | 0x38020c | W | 0x00000000 | DMA Write SAR Low Register. |
| DMA SAR HIGH OFF WR CH_0 | 0x380210 | W | 0x00000000 | DMA Write SAR High Register. |
| DMA DAR LOW OFF WR CH_0 | 0x380214 | W | 0x00000000 | DMA Write DAR Low Register. |
| DMA DAR HIGH OFF WR CH_0 | 0x380218 | W | 0x00000000 | DMA Write DAR High Register. |
| DMA LLP LOW OFF WRC H_0 | 0x38021c | W | 0x00000000 | DMA Write Linked List Pointer Low Register. |
| DMA LLP HIGH OFF WR CH_0 | 0x380220 | W | 0x00000000 | DMA Write Linked List Pointer High Register. |
| DMA CH CONTROL1 OFF RDCH_0 | 0x380300 | W | 0x00000000 | DMA Read Channel Control 1 Register. |
| DMA TRANSFER SIZE OF F_RDCH_0 | 0x380308 | W | 0x00000000 | DMA Read Transfer Size Register. |
| DMA SAR LOW OFF RDC H_0 | 0x38030c | W | 0x00000000 | DMA Read SAR Low Register. |
| DMA SAR HIGH OFF RD CH_0 | 0x380310 | W | 0x00000000 | DMA Read SAR High Register. |
| DMA DAR LOW OFF RD CH_0 | 0x380314 | W | 0x00000000 | DMA Read DAR Low Register. |
| DMA DAR HIGH OFF RD CH_0 | 0x380318 | W | 0x00000000 | DMA Read DAR High Register. |
| DMA LLP LOW OFF RDC H_0 | 0x38031c | W | 0x00000000 | DMA Read Linked List Pointer Low Register. |
| DMA LLP HIGH OFF RDC H_0 | 0x380320 | W | 0x00000000 | DMA Read Linked List Pointer High Register. |

10.5.37 DMA Detail Register Description

DMA CTRL DATA ARB PRIOR OFF

Address: Operational Base + offset (0x380000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:12 | RO | 0x0 | reserved |
| 11:9 | RW | 0x3 | RDBUFF_TRGT_WEIGHT DMA Read Channel MWr Requests. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 8:6 | RW | 0x2 | <p>RD_CTRL_TRGT_WEIGHT DMA Read Channel MRd Requests. For LL element/descriptor access.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 5:3 | RW | 0x1 | <p>WR_CTRL_TRGT_WEIGHT DMA Write Channel MRd Requests. For DMA data requests and LL element/descriptor access.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 2:0 | RW | 0x0 | <p>RTRGT1_WEIGHT Non-DMA Rx Requests.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA_CTRL_OFF

Address: Operational Base + offset (0x380008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:26 | RO | 0x0 | reserved |
| 25 | RW | 0x0 | <p>DIS_C2W_CACHE_RD Disable DMA Read Channels "completion to memory write" context cache pre-fetch function.</p> <p>Note: For internal debugging only.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 24 | RW | 0x0 | <p>DIS_C2W_CACHE_WR Disable DMA Write Channels "completion to memory write" context cache pre-fetch function.</p> <p>Note: For internal debugging only.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 23:20 | RO | 0x0 | reserved |
| 19:16 | RO | 0x1 | <p>NUM_DMA_RD_CHAN Number of Read Channels. You can read this register to determine the number of read channels the DMA controller has been configured to support.</p> |
| 15:4 | RO | 0x0 | reserved |
| 3:0 | RO | 0x1 | <p>NUM_DMA_WR_CHAN Number of Write Channels. You can read this register to determine the number of write channels the DMA controller has been configured to support.</p> |

DMA_WRITE_ENGINE_EN_OFF

Address: Operational Base + offset (0x38000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | <p>DMA_WRITE_ENGINE_EN_HSHAKE_CH7 Enable Handshake for DMA Write Engine Channel 7.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 22 | RW | 0x0 | <p>DMA_WRITE_ENGINE_EN_HSHAKE_CH6 Enable Handshake for DMA Write Engine Channel 6.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 21 | RW | 0x0 | <p>DMA_WRITE_ENGINE_EN_HSHAKE_CH5 Enable Handshake for DMA Write Engine Channel 5.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 20 | RW | 0x0 | DMA_WRITE_ENGINE_EN_HSHAKE_CH4 Enable Handshake for DMA Write Engine Channel 4. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 19 | RW | 0x0 | DMA_WRITE_ENGINE_EN_HSHAKE_CH3 Enable Handshake for DMA Write Engine Channel 3. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 18 | RW | 0x0 | DMA_WRITE_ENGINE_EN_HSHAKE_CH2 Enable Handshake for DMA Write Engine Channel 2. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 17 | RW | 0x0 | DMA_WRITE_ENGINE_EN_HSHAKE_CH1 Enable Handshake for DMA Write Engine Channel 1. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 16 | RW | 0x0 | DMA_WRITE_ENGINE_EN_HSHAKE_CH0 Enable Handshake for DMA Write Engine Channel 0. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 15:1 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 0 | RW | 0x0 | <p>DMA_WRITE_ENGINE DMA Write Engine Enable. 1: Enable 0: Disable (Soft Reset)</p> <p>For normal operation, you must initially set this bit to "1", before any other software setup actions. You do not need to toggle or rewrite to this bit during normal operation. You should set this bit to "0" when you want to "Soft Reset" the DMA controller write logic. There are three possible reasons for resetting the DMA controller write logic:</p> <p>The "Abort Interrupt Status" bit is set (in the "DMA Write Interrupt Status Register" DMA_WRITE_INT_STATUS_OFF), and any of the bits is in the "DMA Write Error Status Register" (DMA_WRITE_ERR_STATUS_OFF) are set. Resetting the DMA controller write logic re-initializes the control logic, ensuring that the next DMA write transfer is executed successfully.</p> <p>You have executed the procedure outlined in "Stop Bit" , after which, the "Abort Interrupt Status" bit is set and the Channel Status field (CS) of the DMA write "DMA Channel Control 1 Register " (DMA_CH_CONTROL1_OFF_WRCH_0) is set to "Stopped." Resetting the DMA controller write logic re-initializes the control logic ensuring that the next DMA write transfer is executed successfully.</p> <p>During software development, when you incorrectly program the DMA write engine.</p> <p>To "Soft Reset" the DMA controller write logic, you must: De-assert the DMA write engine enable bit. Wait for the DMA to complete any in-progress TLP transfer, by waiting until a read on the DMA write engine enable bit returns a "0". Assert the DMA write engine enable bit. This "Soft Reset" does not clear the DMA configuration registers. The DMA write transfer does not start until you write to the "DMA Write Doorbell Register" (DMA_WRITE_DOORBELL_OFF).</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA WRITE DOORBELL OFF

Address: Operational Base + offset (0x380010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | <p>WR_STOP Stop. Set in conjunction with the Doorbell Number field. The DMA write channel stops issuing requests, sets the channel status to "Stopped", and asserts the "Abort" interrupt if it is enabled. Before setting the Stop bit, you must read the channel Status field (CS) of the "DMA Channel Control 1 Register " (DMA_CH_CONTROL1_OFF_WRCH_0) to ensure that the write channel is "Running" (transferring data). For more information, see "Stopping the DMA Transfer (Software Stop)." Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 30:3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x0 | <p>WR_DOORBELL_NUM Doorbell Number. You must write the channel number to this register to start the DMA write transfer for that channel. The DMA detects a write to this register field even if the value of this field does not change. You do not need to toggle or write any other value to this register to start a new transfer. The range of this field is 0x0 to 0x7, and 0x0 corresponds to channel 0. Also note that a write to this field triggers the controller to exit L1 substates. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA_WRITE CHANNEL ARB WEIGHT LOW OFF

Address: Operational Base + offset (0x380018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x01 | <p>WRITE_CHANNEL3_WEIGHT Channel 3 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 14:10 | RW | 0x01 | <p>WRITE_CHANNEL2_WEIGHT Channel 2 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 9:5 | RW | 0x01 | <p>WRITE_CHANNEL1_WEIGHT Channel 1 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 4:0 | RW | 0x01 | <p>WRITE_CHANNEL0_WEIGHT Channel 0 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA WRITE CHANNEL ARB WEIGHT HIGH OFF

Address: Operational Base + offset (0x38001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x01 | <p>WRITE_CHANNEL7_WEIGHT Channel 7 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 14:10 | RW | 0x01 | <p>WRITE_CHANNEL6_WEIGHT Channel 6 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 9:5 | RW | 0x01 | <p>WRITE_CHANNEL5_WEIGHT Channel 5 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 4:0 | RW | 0x01 | <p>WRITE_CHANNEL4_WEIGHT Channel 4 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. A value of '0' means that one TLP is issued before moving to the next channel.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA READ ENGINE EN OFF

Address: Operational Base + offset (0x38002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH7 Enable Handshake for DMA Read Engine Channel 7. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 22 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH6 Enable Handshake for DMA Read Engine Channel 6. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 21 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH5 Enable Handshake for DMA Read Engine Channel 5. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 20 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH4 Enable Handshake for DMA Read Engine Channel 4. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 19 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH3 Enable Handshake for DMA Read Engine Channel 3. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 18 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH2 Enable Handshake for DMA Read Engine Channel 2. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 17 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH1 Enable Handshake for DMA Read Engine Channel 1. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 16 | RW | 0x0 | DMA_READ_ENGINE_EN_HSHAKE_CH0 Enable Handshake for DMA Read Engine Channel 0. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 15:1 | RO | 0x0 | reserved |

| | | | |
|---|----|-----|--|
| | | | DMA_READ_ENGINE DMA Read Engine Enable. 1: Enable 0: Disable (Soft Reset) For normal operation, you must initially set this bit to "1", before any other software setup actions. You do not need to toggle or rewrite to this bit during normal operation. You should set this field to "0" when you want to "Soft Reset" the DMA controller read logic. The "Abort Interrupt Status" bit is set (in the "DMA Read Interrupt Status Register" (DMA_READ_INT_STATUS_OFF), and any of the bits in the "DMA Read Error Status Low Register" (DMA_READ_ERR_STATUS_LOW_OFF) is set. Resetting the DMA controller read logic re-initializes the control logic, ensuring that the next DMA read transfer is executed successfully. You have executed the procedure outlined in "Stop Bit", after which, the "Abort Interrupt Status" bit is set and the channel Status field (CS) of the DMA read "DMA Channel Control 1 Register" (DMA_CH_CONTROL1_OFF_WRCH_0) is set to "Stopped". Resetting the DMA controller read logic re-initializes the control logic ensuring that the next DMA read transfer is executed successfully. During software development, when you incorrectly program the DMA read engine. To "Soft Reset" the DMA controller read logic, you must: De-assert the DMA read engine enable bit. Wait for the DMA to complete any in-progress TLP transfer, by waiting until a read on the DMA read engine enable bit returns a "0". Assert the DMA read engine enable bit. This "Soft Reset" does not clear the DMA configuration registers. The DMA read transfer does not start until you write to the "DMA Read Doorbell Register" (DMA_READ_DOORBELL_OFF). Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 0 | RW | 0x0 | |

DMA READ DOORBELL OFF

Address: Operational Base + offset (0x380030)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31 | RW | 0x0 | <p>RD_STOP Stop. Set in conjunction with the Doorbell Number field. The DMA read channel stops issuing requests, sets the channel status to "Stopped", and asserts the "Abort" interrupt if it is enabled.</p> <p>Before setting the Stop bit, you must read the channel Status field (CS) of the "DMA Channel Control 1 Register" (DMA_CH_CONTROL1_OFF_RDCH_0) to ensure that the read channel is "Running" (transferring data). For more information, see "Stopping the DMA Transfer (Software Stop)".</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 30:3 | RO | 0x0 | reserved |
| 2:0 | RW | 0x0 | <p>RD_DOORBELL_NUM Doorbell Number. You must write 0x0 to this register to start the DMA read transfer for that channel. The DMA detects a write to this register field even if the value of this field does not change.</p> <p>The range of this field is 0x0 to 0x7, and 0x0 corresponds to channel 0. Also note that a write to this field triggers the controller to exit L1 substates.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA READ CHANNEL ARB WEIGHT LOW OFF

Address: Operational Base + offset (0x380038)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x01 | <p>READ_CHANNEL3_WEIGHT Channel 3 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 14:10 | RW | 0x01 | <p>READ_CHANNEL2_WEIGHT Channel 2 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 9:5 | RW | 0x01 | <p>READ_CHANNEL1_WEIGHT Channel 1 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 4:0 | RW | 0x01 | <p>READ_CHANNEL0_WEIGHT Channel 0 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA READ CHANNEL ARB WEIGHT HIGH OFF

Address: Operational Base + offset (0x38003c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:20 | RO | 0x0 | reserved |
| 19:15 | RW | 0x01 | <p>READ_CHANNEL7_WEIGHT Channel 7 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 14:10 | RW | 0x01 | <p>READ_CHANNEL6_WEIGHT Channel 6 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 9:5 | RW | 0x01 | <p>READ_CHANNEL5_WEIGHT Channel 5 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 4:0 | RW | 0x01 | <p>READ_CHANNEL4_WEIGHT Channel 4 Weight. The weight is initialized by software before ringing the doorbell. The value is used by the channel weighted round robin arbiter to select the next channel read request.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA WRITE INT STATUS OFF

Address: Operational Base + offset (0x38004c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RW | 0x00 | <p>WR_ABORT_INT_STATUS Abort Interrupt Status. The DMA write channel has detected an error, or you manually stopped the transfer as described in "Error Handling Assistance by Remote Software". Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA write interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the DMA write interrupt Clear register to clear this interrupt bit.</p> <p>Note: You can write to this register to emulate interrupt generation, during software or hardware testing. A write to the address triggers an interrupt, but the DMA does not set the Done or Abort bits in this register.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 15:8 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 7:0 | RW | 0x00 | <p>WR_DONE_INT_STATUS Done Interrupt Status. The DMA write channel has successfully completed the DMA transfer. For more details, see "Interrupts and Error Handling". Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA write interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the DMA write interrupt Clear register to clear this interrupt bit.</p> <p>Note: You can write to this register to emulate interrupt generation, during software or hardware testing. A write to the address triggers an interrupt, but the DMA does not set the Done or Abort bits in this register.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |

DMA WRITE INT MASK OFF

Address: Operational Base + offset (0x380054)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16 | RW | 0x1 | <p>WR_ABORT_INT_MASK Abort Interrupt Mask. Prevents the Abort interrupt status field in the DMA write interrupt status register from asserting the edma_int output. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |
| 15:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | <p>WR_DONE_INT_MASK Done Interrupt Mask. Prevents the Done interrupt status field in the DMA write interrupt status register from asserting the edma_int output. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |

DMA WRITE INT CLEAR OFF

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|-------------|
| 31:17 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 16 | WO | 0x0 | <p>WR_ABORT_INT_CLEAR Abort Interrupt Clear. You must write a 1'b1 to clear the corresponding bit in the Abort interrupt status field of the DMA write interrupt status register. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: Reading from this self-clearing register field always returns a "0".</p> |
| 15:1 | RO | 0x0 | reserved |
| 0 | WO | 0x0 | <p>WR_DONE_INT_CLEAR Done Interrupt Clear. You must write a 1'b1 to clear the corresponding bit in the Done interrupt status field of the DMA write interrupt status register. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: Reading from this self-clearing register field always returns a "0".</p> |

DMA WRITE ERR STATUS OFF

Address: Operational Base + offset (0x38005c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | RO | 0x00 | <p>LINKLIST_ELEMENT_FETCH_ERR_DETECT Linked List Element Fetch Error Detected. The DMA write channel has received an error response from the AXI bus (or TRGT1 interface when the AXI Bridge is not used) while reading a linked list element from local memory. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling". Masking: The DMA write interrupt Mask register has no effect on this register. Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Write Interrupt Clear Register" (DMA_WRITE_INT_CLEAR_OFF) to clear this error bit. Value After Reset: 0x0</p> |
| 15:8 | RO | 0x0 | reserved |
| 7:0 | RO | 0x00 | <p>APP_READ_ERR_DETECT Application Read Error Detected. The DMA write channel has received an error response from the AXI bus (or TRGT1 interface when the AXI Bridge is not used) while reading data from it. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling". Masking: The DMA write interrupt Mask register has no effect on this register. Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Write Interrupt Clear Register" (DMA_WRITE_INT_CLEAR_OFF) to clear this error bit.</p> |

DMA WRITE DONE IMWR LOW OFF

Address: Operational Base + offset (0x380060)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>DMA_WRITE_DONE_LOW_REG The DMA uses this field to generate bits [31:0] of the address field for the Done IMWr TLP. Bits [1:0] must be "00" as this address must be dword aligned.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA WRITE DONE IMWR HIGH OFF

Address: Operational Base + offset (0x380064)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | DMA_WRITE_DONE_HIGH_REG The DMA uses this field to generate bits [63:32] of the address field for the Done IMWr TLP. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA WRITE ABORT IMWR LOW OFF

Address: Operational Base + offset (0x380068)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | DMA_WRITE_ABORT_LOW_REG The DMA uses this field to generate bits [31:0] of the address field for the Abort IMWr TLP it generates. Bits [1:0] must be "00" as this address must be dword aligned. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA WRITE ABORT IMWR HIGH OFF

Address: Operational Base + offset (0x38006c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | DMA_WRITE_ABORT_HIGH_REG The DMA uses this field to generate bits [63:32] of the address field for the Abort IMWr TLP. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA WRITE CH01 IMWR DATA OFF

Address: Operational Base + offset (0x380070)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:16 | RW | 0x0000 | WR_CHANNEL_1_DATA The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 1. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 15:0 | RW | 0x0000 | WR_CHANNEL_0_DATA The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 0. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA WRITE CH23 IMWR DATA OFF

Address: Operational Base + offset (0x380074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | <p>WR_CHANNEL_3_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 3.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |
| 15:0 | RW | 0x0000 | <p>WR_CHANNEL_2_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 2.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |

DMA WRITE CH45 IMWR DATA OFF

Address: Operational Base + offset (0x380078)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | <p>WR_CHANNEL_5_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 5.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |
| 15:0 | RW | 0x0000 | <p>WR_CHANNEL_4_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 4.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |

DMA WRITE CH67 IMWR DATA OFF

Address: Operational Base + offset (0x38007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | WR_CHANNEL_7_DATA The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 7. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 15:0 | RW | 0x0000 | WR_CHANNEL_6_DATA The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for write channel 6. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA_WRITE_LINKED_LIST_ERR_EN_OFF

Address: Operational Base + offset (0x380090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | WR_CHANNEL_LLLAIE Write Channel LL Local Abort Interrupt Enable (LLLAIE). You enable the write channel local abort interrupt through this bit. The LIE and RIE bits in the LL element enable the write channel done interrupts. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0. Used in linked list mode only. For more details, see "Interrupt Handling". Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 15:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | WR_CHANNEL_LLRAIE Write Channel LL Remote Abort Interrupt Enable (LLRAIE). You enable the write channel remote abort interrupt through this bit. The LIE and RIE bits in the LL element enable the write channel done interrupts. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0. Used in linked list mode only. For more details, see "Interrupt Handling". Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA_READ_INT_STATUS_OFF

Address: Operational Base + offset (0x3800a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:24 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 23:16 | RW | 0x00 | <p>RD_ABORT_INT_STATUS Abort Interrupt Status. The DMA read channel has detected an error, or you manually stopped the transfer as described in "Stopping the DMA Transfer (Software Stop)". Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0. You can read the "DMA Read Error Status Low Register" (DMA_READ_ERR_STATUS_LOW_OFF) and "DMA Read Error Status High Register" (DMA_READ_ERR_STATUS_HIGH_OFF) to determine the source of the error.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA read interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the DMA read interrupt Clear register to clear this interrupt bit.</p> <p>Note: You can write to this register to emulate interrupt generation, during software or hardware testing. A write to the address triggers an interrupt, but the DMA does not set the Done or Abort bits in this register.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |
| 15:8 | RO | 0x0 | reserved |
| 7:0 | RW | 0x00 | <p>RD_DONE_INT_STATUS Done Interrupt Status. The DMA read channel has successfully completed the DMA read transfer. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA read interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the DMA read interrupt Clear register to clear this interrupt bit.</p> <p>Note: You can write to this register to emulate interrupt generation, during software or hardware testing. A write to the address triggers an interrupt, but the DMA does not set the Done or Abort bits in this register.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W</p> <p>Dbi: R/W</p> |

DMA READ INT MASK OFF

Address: Operational Base + offset (0x3800a8)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|-------------|
| 31:17 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 16 | RW | 0x1 | <p>RD_ABORT_INT_MASK Abort Interrupt Mask. Prevents the Abort interrupt status field in the DMA read interrupt status register from asserting the edma_int output. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 15:1 | RO | 0x0 | reserved |
| 0 | RW | 0x1 | <p>RD_DONE_INT_MASK Done Interrupt Mask. Prevents the Done interrupt status field in the DMA read interrupt status register from asserting the edma_int output. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA READ INT CLEAR OFF

Address: Operational Base + offset (0x3800ac)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:24 | RO | 0x0 | reserved |
| 23:16 | WO | 0x00 | <p>RD_ABORT_INT_CLEAR Abort Interrupt Clear. You must write a 1'b1 to clear the corresponding bit in the Abort interrupt status field of the DMA read interrupt status register. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: Reading from this self-clearing register field always returns a "0".</p> |
| 15:8 | RO | 0x0 | reserved |
| 7:0 | WO | 0x00 | <p>RD_DONE_INT_CLEAR Done Interrupt Clear. You must write a 1'b1 to clear the corresponding bit in the Done interrupt status field of the DMA read interrupt status register. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Note: Reading from this self-clearing register field always returns a "0".</p> |

DMA READ ERR STATUS LOW OFF

Address: Operational Base + offset (0x3800b4)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|-------------|
| 31:24 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 23:16 | RO | 0x00 | <p>LINK_LIST_ELEMENT_FETCH_ERR_DETECT Linked List Element Fetch Error Detected. The DMA read channel has received an error response from the AXI bus while reading a linked list element from local memory. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling". Masking: The DMA read interrupt Mask register has no effect on this register. Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Read Interrupt Clear Register" (DMA_READ_INT_CLEAR_OFF) to clear this error bit. Note, this clears all bits in this register, and also the DMA Read Error Status High register (DMA_READ_ERR_STATUS_HIGH_OFF).</p> |
| 15:8 | RO | 0x0 | reserved |
| 7:0 | RO | 0x00 | <p>APP_WR_ERR_DETECT Application Write Error Detected. The DMA read channel has received an error response from the AXI bus (or TRGT1 interface when the AXI Bridge is not used) while writing data to it. This error is fatal. You must restart the transfer from the beginning, as the channel context is corrupted, and the transfer is not rolled back. For more details, see "Linked List Mode". Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling". Masking: The DMA read interrupt Mask register has no effect on this register. Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Read Interrupt Clear Register" (DMA_READ_INT_CLEAR_OFF) to clear this error bit. Note, this clears all bits in this register, and also the DMA Read Error Status High register (DMA_READ_ERR_STATUS_HIGH_OFF).</p> |

DMA_READ_ERR_STATUS_HIGH_OFF

Address: Operational Base + offset (0x3800b8)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:24 | RO | 0x00 | <p>DATA_POISIONING Data Poisoning. The DMA read channel has detected data poisoning in the completion from the remote device (in response to the MRd request). The DMA read channel will drop the completion and then be halted. The CX_FLT_MASK_UR_POIS filter rule does not affect this behavior. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA read interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Read Interrupt Clear Register" (DMA_READ_INT_CLEAR_OFF) to clear this error bit.</p> <p>Note, this also clears the other error bits for the same channel in this register and in the DMA Read Error Status Low register.</p> |
| 23:16 | RO | 0x00 | <p>CPL_TIMEOUT Completion Time Out. The DMA read channel has timed-out while waiting for the remote device to respond to the MRd request, or a malformed CplD has been received. For more details, see "Linked List Mode". Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA read interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Read Interrupt Clear Register" (DMA_READ_INT_CLEAR_OFF) to clear this error bit.</p> <p>Note, this also clears the other error bits for the same channel in this register and in the DMA Read Error Status Low register.</p> |
| 15:8 | RO | 0x00 | <p>CPL_ABORT Completer Abort. The DMA read channel has received a PCIe completer abort completion status from the remote device in response to the MRd request. For more details, see "Linked List Mode". Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA read interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Read Interrupt Clear Register" (DMA_READ_INT_CLEAR_OFF) to clear this error bit.</p> <p>Note, this also clears the other error bits for the same channel in this register and in the DMA Read Error Status Low register.</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 7:0 | RO | 0x00 | <p>UNSUPPORTED_REQ Unsupported Request. The DMA read channel has received a PCIe unsupported request completion status from the remote device in response to the MRd request. For more details, see "Linked List Mode". Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0.</p> <p>Enabling: For details, see "Interrupts and Error Handling".</p> <p>Masking: The DMA read interrupt Mask register has no effect on this register.</p> <p>Clearing: You must write a 1'b1 to the corresponding channel bit in the Abort interrupt field of the "DMA Read Interrupt Clear Register" (DMA_READ_INT_CLEAR_OFF) to clear this error bit.</p> <p>Note, this also clears the other error bits for the same channel in this register and in the DMA Read Error Status Low register.</p> |

DMA READ LINKED LIST ERR EN OFF

Address: Operational Base + offset (0x3800c4)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | <p>RD_CHANNEL_LLLAIE Read Channel LL Local Abort Interrupt Enable (LLLAIE). You enable the read channel Local Abort interrupt through this bit. The LIE and RIE bits in the LL element enable the read channel done interrupts. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0. Used in linked list mode only. For more details, see "Interrupt Handling".</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 15:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | <p>RD_CHANNEL_LLRAIE Read Channel LL Remote Abort Interrupt Enable (LLRAIE). You enable the read channel Remote Abort interrupt through this bit. The LIE and RIE bits in the LL element enable the read channel done interrupts. Each bit corresponds to a DMA channel. Bit [0] corresponds to channel 0. Used in linked list mode only. For more details, see "Interrupt Handling".</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA READ DONE IMWR LOW OFF

Address: Operational Base + offset (0x3800cc)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | DMA_READ_DONE_LOW_REG The DMA uses this field to generate bits [31:0] of the address field for the Done IMWr TLP. Bits [1:0] must be "00" as this address must be dword aligned. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA READ DONE IMWR HIGH OFF

Address: Operational Base + offset (0x3800d0)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | DMA_READ_DONE_HIGH_REG The DMA uses this field to generate bits [63:32] of the address field for the Done IMWr TLP. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA READ ABORT IMWR LOW OFF

Address: Operational Base + offset (0x3800d4)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | DMA_READ_ABORT_LOW_REG The DMA uses this field to generate bits [31:0] of the address field for the Abort IMWr TLP. Bits [1:0] must be "00" as this address must be dword aligned. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA READ ABORT IMWR HIGH OFF

Address: Operational Base + offset (0x3800d8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | DMA_READ_ABORT_HIGH_REG The DMA uses this field to generate bits [63:32] of the address field for the Abort IMWr TLP. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA READ CH01 IMWR DATA OFF

Address: Operational Base + offset (0x3800dc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | <p>RD_CHANNEL_1_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 1.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 15:0 | RW | 0x0000 | <p>RD_CHANNEL_0_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 0.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA READ CH23 IMWR DATA OFF

Address: Operational Base + offset (0x3800e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | <p>RD_CHANNEL_3_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 3.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 15:0 | RW | 0x0000 | <p>RD_CHANNEL_2_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 2.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA READ CH45 IMWR DATA OFF

Address: Operational Base + offset (0x3800e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | <p>RD_CHANNEL_5_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 5.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 15:0 | RW | 0x0000 | <p>RD_CHANNEL_4_DATA</p> <p>The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 4.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA READ CH67 IMWR DATA OFF

Address: Operational Base + offset (0x3800e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | RD_CHANNEL_7_DATA The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 7. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 15:0 | RW | 0x0000 | RD_CHANNEL_6_DATA The DMA uses this field to generate the data field for the Done or Abort IMWr TLPs it generates for read channel 6. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA WRITE ENGINE HSHAKE CNT LOW OFF

Address: Operational Base + offset (0x380108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28:24 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH3 DMA handshake counter for DMA Write Engine Channel 3. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH2 DMA handshake counter for DMA Write Engine Channel 2. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH1 DMA handshake counter for DMA Write Engine Channel 1. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH0 DMA handshake counter for DMA Write Engine Channel 0. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |

DMA WRITE ENGINE HSHAKE CNT HIGH OFF

Address: Operational Base + offset (0x38010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:29 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 28:24 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH7 DMA handshake counter for DMA Write Engine Channel 7. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH6 DMA handshake counter for DMA Write Engine Channel 6. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH5 DMA handshake counter for DMA Write Engine Channel 5. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RO | 0x00 | DMA_WRITE_ENGINE_HSHAKE_CNT_CH4 DMA handshake counter for DMA Write Engine Channel 4. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |

DMA READ ENGINE HSHAKE CNT LOW OFF

Address: Operational Base + offset (0x380118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28:24 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH4 DMA handshake counter for DMA Read Engine Channel 3. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH2 DMA handshake counter for DMA Read Engine Channel 2. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH1 DMA handshake counter for DMA Read Engine Channel 1. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH0 DMA handshake counter for DMA Read Engine Channel 0. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |

DMA READ ENGINE HSHAKE CNT HIGH OFF

Address: Operational Base + offset (0x38011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:29 | RO | 0x0 | reserved |
| 28:24 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH7 DMA handshake counter for DMA Read Engine Channel 7. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 23:21 | RO | 0x0 | reserved |
| 20:16 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH6 DMA handshake counter for DMA Read Engine Channel 6. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 15:13 | RO | 0x0 | reserved |
| 12:8 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH5 DMA handshake counter for DMA Read Engine Channel 5. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |
| 7:5 | RO | 0x0 | reserved |
| 4:0 | RO | 0x00 | DMA_READ_ENGINE_HSHAKE_CNT_CH4 DMA handshake counter for DMA Read Engine Channel 4. If CC_DMA_HSHAKE =1, the data transfer in Linked List mode starts only when the counter is non-zero. |

DMA CH CONTROL1 OFF WRCH_0

Address: Operational Base + offset (0x380200)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RW | 0x0 | DMA_AT Address Translation TLP Header Bit (AT) The DMA uses this TLP header field when generating MRd/MWr (not IMWr) TLPs. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 29:27 | RW | 0x0 | DMA_TC Traffic Class TLP Header Bit (TC) The DMA uses this TLP header field when generating MRd/MWr (not IMWr) TLPs. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 26 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 25 | RW | 0x0 | DMA_RO Relaxed Ordering TLP Header Bit (RO) The DMA uses this TLP header field when generating MRd/MWr (not IMWr) TLPs. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 24 | RW | 0x0 | DMA_NS_SRC Source No Snoop TLP Header Bit (DMA_NS_SRC). The DMA uses this TLP header field when generating MRd (SAR addressing space) (not IMWr) TLPs. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 23 | RW | 0x0 | DMA_NS_DST Destination No Snoop TLP Header Bit (DMA_NS_DST). The DMA uses this TLP header field when generating MWr (DAR addressing space) (not IMWr) TLPs. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 22:17 | RO | 0x0 | reserved |
| 16:12 | RW | 0x00 | DMA_FUNC_NUM Function Number (FN). The controller uses this field when generating the requester ID for the MRd/MWr DMA TLP. When you have enabled SR-IOV, then this field is ignored if you have set the VFE field in the "DMA Write Channel Control 2 Register" (DMA_CH_CONTROL2_OFF_WRCH_0). Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |
| 11:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | LLE Linked List Enable (LLE). 0: Disable linked list operation 1: Enable linked list operation Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 8 | RW | 0x0 | <p>CCS Consumer Cycle State (CCS). Used in linked list mode only. It is used to synchronize the producer (software) and the consumer (DMA). For more details, see "PCS-CCS-CB-TCB Producer-Consumer Synchronization".</p> <p>You must initialize this bit. The DMA updates this bit during linked list operation.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 7 | RO | 0x0 | reserved |
| 6:5 | RO | 0x0 | <p>CS Channel Status (CS). The channel status bits identify the current operational state of the DMA channel. The operation state encoding for each DMA channel is as follows:</p> <p>00: Reserved 01: Running. This channel is active and transferring data. 10: Halted. An error condition has been detected, and the DMA has stopped this channel. 11: Stopped. The DMA has transferred all data for this channel, or you have prematurely stopped this channel by writing to the Stop field of the "DMA Write Doorbell Register" (DMA_WRITE_DOORBELL_OFF) or "DMA Read Doorbell Register" (DMA_READ_DOORBELL_OFF).</p> |
| 4 | RW | 0x0 | <p>RIE Remote Interrupt Enable (RIE). You must set this bit to enable the generation of the Done or Abort Remote interrupts. For more details, see "Interrupts and Error Handling". In LL mode, the DMA overwrites this with the RIE of the LL element. The RIE bit in a LL element only enables the Done interrupt. In non-LL mode, the RIE bit enables the Done and Abort interrupts.</p> <p>This field is not defined in a link LL element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 3 | RW | 0x0 | <p>LIE Local Interrupt Enable (LIE). You must set this bit to enable the generation of the Done or Abort Local interrupts. For more details, see "Interrupts and Error Handling". In LL mode, the DMA overwrites this with the LIE of the LL element. The LIE bit in a LL element only enables the Done interrupt. In non-LL mode, the LIE bit enables the Done and Abort interrupts.</p> <p>This field is not defined in a link LL element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 2 | RW | 0x0 | <p>LLP Load Link Pointer (LLP). Used in linked list mode only. Indicates that this linked list element is a link element, and its LL element pointer dwords are pointing to the next (non-contiguous) element. The DMA loads this field with the LLP of the linked list element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 1 | RW | 0x0 | <p>TCB Toggle Cycle Bit (TCB). Indicates to the DMA to toggle its interpretation of the CB. Used in linked list mode only. It is used to synchronize the producer (software) and the consumer (DMA). For more details, see "PCS-CCS-CB-TCB Producer-Consumer Synchronization". The DMA loads this field with the TCB of the linked list element. this field is not defined in a data LL element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 0 | RW | 0x0 | <p>CB Cycle Bit (CB). Used in linked list mode only. It is used to synchronize the producer (software) and the consumer (DMA). For more details, see "PCS-CCS-CB-TCB Producer-Consumer Synchronization". The DMA loads this field with the CB of the linked list element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA TRANSFER SIZE OFF WRCH_0

Address: Operational Base + offset (0x380208)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>DMA_TRANSFER_SIZE DMA Transfer Size. You program this register with the size of the DMA transfer. The maximum DMA transfer size is 4Gbytes. The minimum transfer size is one byte (0x1). This field is automatically decremented by the DMA as the DMA write channel transfer progresses. This field indicates the number bytes remaining to be transferred. When all bytes are successfully transferred the current transfer size is zero. In LL mode, the DMA overwrites this register with the corresponding dword of the LL element.</p> <p>You can read this register to monitor the transfer progress, however in some scenarios this register is updated after a delay. For example, when less than 3 channels are doorbelled, this register is updated only after a descriptor finishes(linked list mode), or the transfer ends (non-linked list mode).</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA SAR LOW OFF WRCH 0

Address: Operational Base + offset (0x38020c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>SRC_ADDR_REG_LOW Source Address Register (Lower 32 bits). Indicates the next address to be read from. The DMA increments the SAR as the DMA transfer progresses. In LL mode, the DMA overwrites this with the corresponding dword of the LL element.</p> <p>DMA Read: The SAR is the address of the remote memory. DMA Write: The SAR is the address of the local memory.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA SAR HIGH OFF WRCH 0

Address: Operational Base + offset (0x380210)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>SRC_ADDR_REG_HIGH Source Address Register (Higher 32 bits). In LL mode, the DMA overwrites this with the corresponding dword of the LL element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA DAR LOW OFF WRCH 0

Address: Operational Base + offset (0x380214)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>DST_ADDR_REG_LOW Destination Address Register (Lower 32 bits). Indicates the next address to be written to. The DMA increments the DAR as the DMA transfer progresses. In LL mode, the DMA overwrites this with the corresponding dword of the LL element.</p> <p>DMA Read: The DAR is the address of the local memory. DMA Write: The DAR is the address of the remote memory.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA DAR HIGH OFF WRCH 0

Address: Operational Base + offset (0x380218)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>DST_ADDR_REG_HIGH Destination Address Register (Higher 32 bits). In LL mode, the DMA overwrites this with the corresponding dword of the LL element.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA LLP LOW OFF WRCH 0

Address: Operational Base + offset (0x38021c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>LLP_LOW Lower bits of the address of the linked list transfer list in local memory. Used in linked list mode only. Updated by the DMA to point to the next element in the transfer list after the previous element is consumed.</p> <p>When the current element is a data element; this field is incremented by 6 DWORDS.</p> <p>When the current element is a link element; this field is overwritten by the LL Element Pointer of the element.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA LLP HIGH OFF WRCH 0

Address: Operational Base + offset (0x380220)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>LLP_HIGH Upper 32 bits of the address of the linked list transfer list in local memory. Used in linked list mode only. Updated by the DMA to point to the next element in the transfer list as elements are consumed.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA CH CONTROL1 OFF RDCH_0

Address: Operational Base + offset (0x380300)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:30 | RW | 0x0 | <p>DMA_AT Address Translation TLP Header Bit (AT) The DMA uses this TLP header field when generating MRd/MWr (not IMWr) TLPs.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 29:27 | RW | 0x0 | <p>DMA_TC Traffic Class TLP Header Bit (TC) The DMA uses this TLP header field when generating MRd/MWr (not IMWr) TLPs.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 26 | RO | 0x0 | reserved |
| 25 | RW | 0x0 | <p>DMA_RO Relaxed Ordering TLP Header Bit (RO) The DMA uses this TLP header field when generating MRd/MWr (not IMWr) TLPs.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 24 | RW | 0x0 | <p>DMA_NS_SRC Source No Snoop TLP Header Bit (DMA_NS_SRC). The DMA uses this TLP header field when generating MRd (SAR addressing space) (not IMWr) TLPs.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 23 | RW | 0x0 | <p>DMA_NS_DST Destination No Snoop TLP Header Bit (DMA_NS_DST). The DMA uses this TLP header field when generating MWr (DAR addressing space) (not IMWr) TLPS. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 22:17 | RO | 0x0 | reserved |
| 16:12 | RW | 0x00 | <p>DMA_FUNC_NUM Function Number (FN). The controller uses this field when generating the requester ID for the MRd/MWr DMA TLP. When you have enabled SR-IOV, then this field is ignored if you have set the VFE field in the "DMA Read Channel Control 2 Register" (DMA_CH_CONTROL2_OFF_RDCH_0). Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 11:10 | RO | 0x0 | reserved |
| 9 | RW | 0x0 | <p>LLE Linked List Enable (LLE). 0: Disable linked list operation 1: Enable linked list operation Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 8 | RW | 0x0 | <p>CCS Consumer Cycle State (CCS). Used in linked list mode only. It is used to synchronize the producer (software) and the consumer (DMA). For more details, see "PCS-CCS-CB-TCB Producer-Consumer Synchronization". You must initialize this bit. The DMA updates this bit during linked list operation. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |
| 7 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 6:5 | RO | 0x0 | <p>CS</p> <p>Channel Status (CS). The channel status bits identify the current operational state of the DMA channel. The operation state encoding for each DMA channel is as follows:</p> <ul style="list-style-type: none"> 00: Reserved 01: Running. This channel is active and transferring data. 10: Halted. An error condition has been detected, and the DMA has stopped this channel. 11: Stopped. The DMA has transferred all data for this channel, or you have prematurely stopped this channel by writing to the Stop field of the "DMA Read Doorbell Register" (DMA_WRITE_DOORBELL_OFF) or "DMA Read Doorbell Register" (DMA_READ_DOORBELL_OFF). |
| 4 | RW | 0x0 | <p>RIE</p> <p>Remote Interrupt Enable (RIE). You must set this bit to enable the generation of the Done or Abort Remote interrupts. For more details, see "Interrupts and Error Handling". In LL mode, the DMA overwrites this with the RIE of the LL element. The RIE bit in a LL element only enables the Done interrupt. In non-LL mode, the RIE bit enables the Done and Abort interrupts.</p> <p>This field is not defined in a link LL element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 3 | RW | 0x0 | <p>LIE</p> <p>Local Interrupt Enable (LIE). You must set this bit to enable the generation of the Done or Abort Local interrupts. For more details, see "Interrupts and Error Handling". In LL mode, the DMA overwrites this with the LIE of the LL element. The LIE bit in a LL element only enables the Done interrupt. In non-LL mode, the LIE bit enables the Done and Abort interrupts.</p> <p>This field is not defined in a link LL element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 2 | RW | 0x0 | <p>LLP</p> <p>Load Link Pointer (LLP). Used in linked list mode only. Indicates that this linked list element is a link element, and its LL element pointer dwords are pointing to the next (non-contiguous) element. The DMA loads this field with the LLP of the linked list element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 1 | RW | 0x0 | <p>TCB Toggle Cycle Bit (TCB). Indicates to the DMA to toggle its interpretation of the CB. Used in linked list mode only. It is used to synchronize the producer (software) and the consumer (DMA). For more details, see "PCS-CCS-CB-TCB Producer-Consumer Synchronization". The DMA loads this field with the TCB of the linked list element. this field is not defined in a data LL element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |
| 0 | RW | 0x0 | <p>CB Cycle Bit (CB). Used in linked list mode only. It is used to synchronize the producer (software) and the consumer (DMA). For more details, see "PCS-CCS-CB-TCB Producer-Consumer Synchronization". The DMA loads this field with the CB of the linked list element.</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA TRANSFER SIZE OFF RDCH 0

Address: Operational Base + offset (0x380308)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>DMA_TRANSFER_SIZE DMA Transfer Size. You program this register with the size of the DMA transfer. The maximum DMA transfer size is 4Gbytes. The minimum transfer size is one byte (0x1). This field is automatically decremented by the DMA as the DMA read channel transfer progresses. This field indicates the number bytes remaining to be transferred. When all bytes are successfully transferred the current transfer size is zero. In LL mode, the DMA overwrites this register with the corresponding dword of the LL element.</p> <p>You can read this register to monitor the transfer progress, however in some scenarios this register is updated after a delay. For example, when less than 3 channels are doorbelled, this register is updated only after a descriptor finishes(linked list mode), or the transfer ends (non-linked list mode).</p> <p>Note: The access attributes of this field are as follows:</p> <p>Wire: R/W Dbi: R/W</p> |

DMA SAR LOW OFF RDCH 0

Address: Operational Base + offset (0x38030c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>SRC_ADDR_REG_LOW Source Address Register (Lower 32 bits). Indicates the next address to be read from. The DMA increments the SAR as the DMA transfer progresses. In LL mode, the DMA overwrites this with the corresponding dword of the LL element.</p> <p>DMA Read: The SAR is the address of the remote memory. DMA Read: The SAR is the address of the local memory.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA SAR HIGH OFF RDCH 0

Address: Operational Base + offset (0x380310)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | <p>SRC_ADDR_REG_HIGH Source Address Register (Higher 32 bits). In LL mode, the DMA overwrites this with the corresponding dword of the LL element.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA DAR LOW OFF RDCH 0

Address: Operational Base + offset (0x380314)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | <p>DST_ADDR_REG_LOW Destination Address Register (Lower 32 bits). Indicates the next address to be written to. The DMA increments the DAR as the DMA transfer progresses. In LL mode, the DMA overwrites this with the corresponding dword of the LL element.</p> <p>DMA Read: The DAR is the address of the local memory. DMA Read: The DAR is the address of the remote memory.</p> <p>Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W</p> |

DMA DAR HIGH OFF RDCH 0

Address: Operational Base + offset (0x380318)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | DST_ADDR_REG_HIGH Destination Address Register (Higher 32 bits). In LL mode, the DMA overwrites this with the corresponding dword of the LL element. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA LLP LOW OFF RDCH 0

Address: Operational Base + offset (0x38031c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:0 | RW | 0x00000000 | LLP_LOW Lower bits of the address of the linked list transfer list in local memory. Used in linked list mode only. Updated by the DMA to point to the next element in the transfer list after the previous element is consumed. When the current element is a data element; this field is incremented by 6 DWORDS. When the current element is a link element; this field is overwritten by the LL Element Pointer of the element. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

DMA LLP HIGH OFF RDCH 0

Address: Operational Base + offset (0x380320)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:0 | RW | 0x00000000 | LLP_HIGH Upper 32 bits of the address of the linked list transfer list in local memory. Used in linked list mode only. Updated by the DMA to point to the next element in the transfer list as elements are consumed. Note: The access attributes of this field are as follows: Wire: R/W Dbi: R/W |

10.5.38 PCIe2/USB3 PHY Detail Register Description**TX PLL Registers**

| Offset | Bit | Reset Value | Description |
|--------|-----|-------------|--|
| 0x2020 | 5:0 | 0x6 | TX PLL div, denoted as N |
| 0x2118 | 3:0 | 0x2 | M[11:8] |
| 0x211c | 7:0 | 0x71 | M[7:0] |
| 0x212c | 0 | 0x0 | 1'b0:PLL control the by PIPE interface signal pipe_pd 1'b1:PLL always running |

Notes: Transmitter data rate is calculated as $(2 * Fref * M)/N$, where N is no less than 1

RX CDR Registers

| Offset | Bit | Reset Value | Description | |
|---------------|------------|--------------------|--------------------|--|
| 0x2028 | 7:0 | 0x71 | P[7:0] | Lane 0 RX CDR frequency multiplier factor, denoted as P0 |
| 0x2020 | 7:6 | 0x2 | P[9:8] | |
| 0x2030 | 4:0 | 0x6 | | Lane 0 RX CDR div, denoted as Q0 |
| 0x3028 | 7:0 | 0x71 | P[7:0] | Lane 1 RX CDR frequency multiplier factor, denoted as P1 |
| 0x3020 | 7:6 | 0x2 | P[9:8] | |
| 0x3030 | 4:0 | 0x6 | | Lane 1 RX CDR div, denoted as Q1 |

Notes: Lane 0 receiver data rate is calculated as $(2 * Fref * P0)/Q0$, where Q0 is no less than 1. Lane 1 receiver data rate is calculated as $(2 * Fref * P1)/Q1$, where Q1 is no less than 1

TX and RX Registers

| Offset | Bit | Reset Value | Description |
|---------------|------------|--------------------|--|
| 0x2088 | 7:5 | 0x1 | Function reserved |
| | 4 | 0x1 | lane 0 TX driver pre-emphasize enable, active high 1'b0: pre-emphasize off 1'b1: pre-emphasize on |
| | 3:0 | 0x3 | Function reserved |
| 0x3088 | 7:5 | 0x1 | Function reserved |
| | 4 | 0x1 | lane 1 TX driver pre-emphasize enable, active high 1'b0: pre-emphasize off 1'b1: pre-emphasize on |
| | 3:0 | 0x3 | Function reserved |
| 0x21a8 | 7 | 0x0 | lane 0 TX LDO reference voltage selection 1'b0: reference coming from Band-gap 1'b1: reference from analog power rail |
| | 6:5 | 0x0 | TX LDO voltage tuning bits ,only available when 0x1a8[7] was set to "1" 2'b00: 0.8v from bang-gap 2'b01:0.75v from bang-gap 2'b10:0.825v from bang-gap 2'b11:0.850v from bang-gap |
| | 4:3 | 0x0 | Reserved |
| | 2 | 0x0 | PCIe device mode enable, active high. In USB3.0 mode, this bit should maintain LOW. 1'b0: PCI Express Root Complex 1'b1: PCI Express Endpoint |
| | 1 | 0x0 | PCIe PLL and CDR reference clock source selection 1'b0: reference clock from internal single ended input stage, sinking clock from pllrefclkin 1'b1: reference clock from external differential pair |

| Offset | Bit | Reset Value | Description |
|---------------|------------|--------------------|--|
| | 0 | 0x1 | PCI Express Host 100MHz differential clock output at REFCLKP/N pins enable, active Low, only available when PCIe was in RC mode and 0x21a8[2] was set to "0". 1'b1: disable 100MHz clock output 1'b0: enable 100MHz clock output |
| 0x31a8 | 7 | 0x0 | lane1 TX LDO reference voltage selection 1'b0: reference coming from Band-gap 1'b1: reference from analog power rail |
| | 6:0 | 0x1 | Reserved |
| 0x21ac | 7:6 | 0x0 | Reserved |
| | 5:4 | 0x0 | TX driver output common mode voltage tuning bits, these bits will tune the common mode voltage of TXP/TXN differential signals. These bits apply to both lane0 and lane1 2'b00:550mv 2'b01:525mv 2'b10:500mv 2'b11:600mv |
| | 3:0 | 0x7 | Reserved |
| | 7:5 | 0x1 | Reserved |
| 0x20e0 | 4 | 0x0 | SERDES life clock power down control, active high 1'b0: normal operation mode 1'b1: power down mode, internal life clock stop toggling |
| | 3:0 | 0x0 | Reserved |
| | 7:5 | 0x0 | Reserved |
| 0x2044 | 4 | 0x0 | Band-gap power down control, only will be asserted in whole chip totally power down mode to save power, in this mode PHY will not respond properly, active high 1'b0: normal operation mode 1'b1: power down mode |
| | 3:0 | 0x0 | Reserved |
| | 7:4 | 0x6 | lane 0 TX driver swing tuning bits with weight, "1111" represents the largest swing and "0000" the smallest |
| 0x21b8 | 3:2 | 0x2 | Reserved |
| | 1:0 | 0x1 | lane 0 TX driver pre-emphasize strength tuning bits with weight, "11" represents the strongest and "00" the weakest. Only available when 0x2088[4] was set to "1" |
| | 7:4 | 0x6 | lane 1 TX driver swing tuning bits with weight, "1111" represents the largest swing and "0000" the smallest |
| 0x31b8 | 3:2 | 0x2 | Reserved |
| | 1:0 | 0x1 | lane 1 TX driver pre-emphasize strength tuning bits with weight, "11" represents the strongest and "00" the weakest. Only available when 0x3088[4] was set to "1" |
| 0x2058 | 7:3 | 0x5 | Reserved |

| Offset | Bit | Reset Value | Description |
|---------------|------------|--------------------|---|
| | 2 | 0x0 | lane 0 RX CTLE enable, active low 1'b1: turn off RX equalizer 1'b0: normal operation mode with CTLE on |
| | 1:0 | 0x0 | Reserved |
| 0x3058 | 7:3 | 0x5 | Reserved |
| | 2 | 0x0 | lane 1 RX CTLE enable, active low 1'b1: turn off RX equalizer 1'b0: normal operation mode with CTLE on |
| | 1:0 | 0x0 | Reserved |
| 0x2064 | 7 | 0x0 | Reserved |
| | 6:4 | 0x4 | lane 0 RX equalizer tail current control |
| | 3:0 | 0x9 | Reserved |
| 0x3064 | 7 | 0x0 | Reserved |
| | 6:4 | 0x4 | lane 1 RX equalizer tail current control |
| | 3:0 | 0x9 | Reserved |
| 0x2068 | 7:4 | 0x7 | Reserved |
| | 3:0 | 0x7 | lane 0 RX CTLE tuning bits, more "1" will give higher peak gain in high frequency and more "0" lower high frequency gain. It is recommended that to tune these bits when the input signal integrity was not good |
| 0x3068 | 7:4 | 0x7 | Reserved |
| | 3:0 | 0x7 | lane 1 RX CTLE tuning bits, more "1" will give higher peak gain in high frequency and more "0" lower high frequency gain. It is recommended that to tune these bits when the input signal integrity was not good |
| 0x20d0 | 7:4 | 0x0 | Reserved |
| | 3:0 | 0x0 | Adjust lane 0 CDR phase tracking speed, track speed increase with the register value, slower tracking speed will increase JTL in high frequency but decrease the low frequency region |
| 0x30d0 | 7:4 | 0x0 | Reserved |
| | 3:0 | 0x0 | Adjust lane 1 CDR phase tracking speed, track speed increase with the register value, slower tracking speed will increase JTL in high frequency but decrease the low frequency region |
| 0x2150 | 7:4 | 0x7 | Reserved |
| | 3:0 | 0x2 | lane 0 CDR charge pump current adjust, charge pump current increase when register value increase, smaller charge pump current will decrease the corner frequency and CDR bandwidth |
| 0x3150 | 7:4 | 0x7 | Reserved |
| | 3:0 | 0x2 | lane 1 CDR charge pump current adjust, charge pump current increase when register value increase, smaller charge pump current will decrease the corner frequency and CDR bandwidth |
| 0x215c | 7:5 | 0x4 | lane 0 RX voltage common mode generator strength tuning bits, strength increase with the bit value |

| Offset | Bit | Reset Value | Description |
|---------------|------------|--------------------|--|
| | 4:0 | 0x0 | Reserved |
| 0x315c | 7:5 | 0x4 | lane 1 RX voltage common mode generator strength tuning bits, strength increase with the bit value |
| | 4:0 | 0x0 | Reserved |
| 0x20c8 | 7:4 | 0x1 | Reserved |
| | 3:1 | 0x2 | lane 0, reduce the charge pump bias current to half, active high, charge pump are generated by this bias current. 1: reduce the charge pump bias current to half of original 0: normal charge pump bias current |
| | 0 | 0x0 | Reserved |
| 0x30c8 | 7:4 | 0x1 | Reserved |
| | 3:1 | 0x2 | lane 1, reduce the charge pump bias current to half, active high, charge pump are generated by this bias current 1: reduce the charge pump bias current to half of original 0: normal charge pump bias current |
| | 0 | 0x0 | Reserved |
| 0x2190 | 7:6 | 0x1 | Reserved |
| | 5:4 | 0x0 | TX detect RX reference tuning bits, lower reference voltage will increase the detection sensitivity. Applies to both lane0 and lane1 2'b00:550mv 2'b01:525mv 2'b10:575mv 2'b11:600mv |
| | 3:2 | 0x0 | Input reference voltage of unit gain buffer to create the bias current used in TX driver and RXEQ. Higher voltage reference will increase the bias current for the PHY. Applies to both lane 0 and lane1 2'b00:600mv 2'b01:575mv 2'b10:625mv 2'b11:650mv |
| | 1:0 | 0x0 | Reserved |
| 0x21a0 | 7:6 | 0x0 | RX electrical idle detection threshold tuning reference, positive input of the threshold comparator, defined as vref1 2'b00:250mv 2'b01:225mv 2'b10:300mv 2'b11:350mv |
| | 5:4 | 0x0 | RX electrical idle detection threshold tuning reference, positive input of the threshold comparator, defined as vref2 2'b00:200mv 2'b01:150mv 2'b10:175mv 2'b11:225mv |

| Offset | Bit | Reset Value | Description |
|---------------|------------|--------------------|--|
| 0x2188 | 3:2 | 0x0 | RX electrical idle detection threshold tuning reference, negative input of the threshold comparator, defined as vref3 2'b00:175mv 2'b01:150mv 2'b10:200mv 2'b11:225mv |
| | 1:0 | 0x0 | RX electrical idle detection threshold tuning reference, negative input of the threshold comparator, defined as vref4 2'b00:75mv 2'b01:50mv 2'b10:100mv 2'b11:125mv |
| 0x3188 | 7:2 | 0x0 | Reserved |
| | 1 | 0x1 | lane 0 RX electrical idle detection threshold tuning reference, positive input selection between vref1 and vref2 defined above, output was defined as verfp* 1'b0:Vref1 1'b1:Vref2 |
| | 0 | 0x0 | lane0 RX electrical idle detection threshold tuning reference, negative input selection between vref3 and vref4 defined above, output was defined as verfn* |
| 0x206c | 7:2 | 0x0 | Reserved |
| | 1 | 0x1 | lane1 RX electrical idle detection threshold tuning reference, positive input selection between vref1 and vref2 defined above, output was defined as verfp* 1'b0:Vref1 1'b1:Vref2 |
| | 0 | 0x0 | lane1 RX electrical idle detection threshold tuning reference, negative input selection between vref3 and vref4 defined above, output was defined as verfn* |
| 0x2070 | 3:0 | 0xf | RX ODT tuning bits, more "1" represents smaller RX termination resistors and more "0" represents larger value. |
| 0x2074 | 7:0 | 0xff | |
| 0x2074 | 7:0 | 0x0 | These bits will only can be over write through registers when 0x2258[3] was forced by '1' |
| 0x2098 | 3:0 | 0xf | TX ODT tuning bits, more "1" represents smaller TX termination resistors and more "0" represents larger value. |
| 0x209c | 7:0 | 0xff | |
| 0x20a0 | 7:0 | 0xc0 | These bits will only can be over write through registers when 0x2258[3] was forced by '1' |
| 0x2258 | 7:4 | 0x0 | Reserved |
| | 3 | 0x0 | TX and RX ODT setting bypass enable 1'b0:not bypassed 1'b1:bypassed |
| | 2:0 | 0x0 | Reserved |
| 0x21ec | 7:2 | 0x2 | Reserved |

| Offset | Bit | Reset Value | Description |
|--------|-----|-------------|---|
| | 1 | 0x0 | Reference Clock output select: 1'b1: use TX PLL clock as REFCLK output 1'b0: use Internal single ended reference clock as REFCLK output |
| | 0 | 0x0 | Reserved |

Notes: The final RX electrical idle detection threshold will be defined as $|vrefp-vrefn|$

SERDES PHY Elasticity Buffer Setting Registers

| Offset | Bit | Reset Value | Description |
|--------|-----|-------------|--|
| 0x45c0 | 7:2 | 0x0 | Reserved |
| | 1:0 | 0x1 | S0P[9:8] |
| 0x45c4 | 7:0 | 0x83 | S0P[7:0] |
| | | | USB3.0 mode:10'h183 PCI Express mode:10'h283 |
| 0x45c8 | 7:2 | 0x0 | Reserved |
| | 1:0 | 0x1 | S1P[9:8] |
| 0x45cc | 7:0 | 0x83 | S1P[7:0] |
| | | | USB3.0 mode:10'h183(default) PCI Express mode:10'h343 |
| 0x45d0 | 7:2 | 0x0 | Reserved |
| | 1:0 | 0x2 | S1N[9:8] |
| 0x45d4 | 7:0 | 0x7c | S1N[7:0] |
| | | | Negative encoding of symbol1 for elasticity buffer USB3.0 mode:10'h27c(default) PCI Express mode:10'h0bc |

BIST Error Counter and Clear Bits

| Offset | Bit | Reset Value | Description |
|--------|-----|-------------|--|
| 0x41e0 | 7:3 | 0x2 | Reserved |
| | 2 | 0x0 | Clear BIST error counter on lane0, active HIGH. This bit should be programmed as zero prior to assertion of 'bist_en' |
| 0x4304 | 1:0 | 0x0 | Reserved |
| | 7:0 | RO | Error[15:8] |
| 0x4308 | 7:0 | | Error[7:0] |
| 0x42cc | 7:0 | 0x80 | NUM[7:0] |
| | | | BIST data length, the number of random data in UI is calculated as '(NUM+1)*20' |

10.6 Application Notes

10.6.1 Clock and Reset

10.6.1.1 Clock Overview

The PCIe module uses multiple clock domains, it consists of system bus clocks (APB and AXI), Core function clock, PIPE interface clock, Power management clock, and PCIe PHY reference clock.

The system bus clock is generated from SOC main PLL internally, please refer to the relative chapter for details. System bus clock gating is handled automatically by PCIe

controller, and the clock gating can be disable by software through PCIE_CLIENT_LOCAL_CRU_CTRL Register.

Core function clock is derived from the PIPE interface clock output from the PCIe PHY. In specific low power state, core function clock is gated and switch to Power management clock automatically.

The PHY TX PLL generates PIPE interface clock from the platform reference clock. The PCIe link Training and transaction is completely dependent upon availability of this clock.

Application should not initiate transactions before ensuring that the PHY PLL is locked.

The Power management clock which should be fixed with 24MHz driven by crystal input clock. The PCIe controller uses this clock for counting time during L1 substates. You must program the frequency of this clock into the L1_SUBSTATES_OFF register with a value of 24MHz to count real time.

Many PCIe connections, especially backplane connections, require a synchronous reference clock between the two link partners. To achieve this a common clock source, referred to as REFCLK in the PCI Express Card Electromechanical Specification, should be used by both ends of the PCIe link. the PCIe PHY provides 100MHz differential clock output (optional with SSC) in RC mode for system application. If Spread Spectrum Clocking (SSC) is used it is required that a common reference clock be used by the link partners. Most commercially available platforms with PCIe backplanes use Spread Spectrum Clocking to reduce EMI. If common clock architecture is used, the user driver should configure the PHY internal configuration bits to enable the feature of 100MHz differential clock output.

10.6.1.2 PCIe PHY Reference Clock

In RC mode, PCIe PHY use internal single-ended clock which can be configured to 24Mhz, 25Mhz, 50Mhz or 100Mhz. Meanwhile, PCIe PHY can provide differential clock output for system application. The source of differential clock output can be select from internal single-ended clock or PHY TX PLL.

In EP mode, PCIe PHY can use internal single-ended clock or external differential clock input as its reference clock. The Reference clock structure and related configuration is shown below.

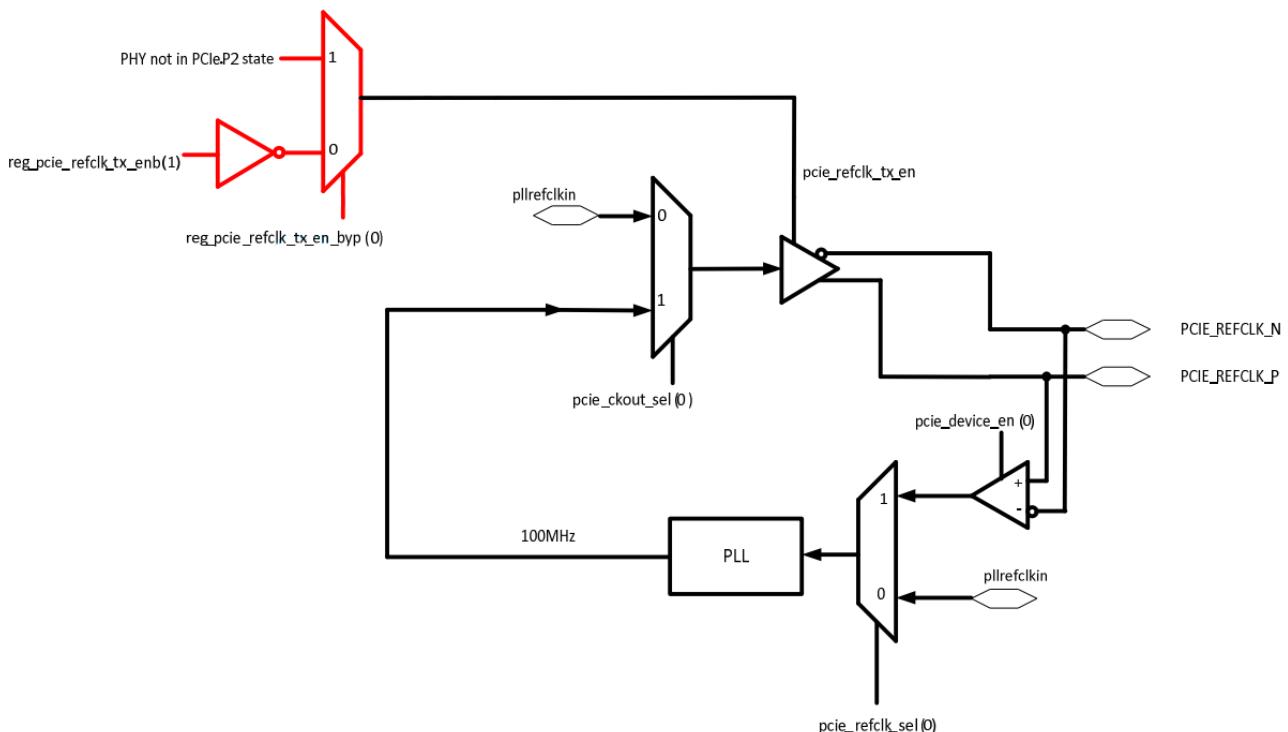


Fig. 10-4 PCIe PHY Reference Clock Structure

Signal pllrefclkin represents internal single-ended clock and PCIE_REFCLK_P/N represent external differential clock. Other select or enable signals can be configured by PCIe PHY register group:

Table 10-5 PCIe PHY reference clock registers

| Offset | Bit | Reset Value | Description |
|---------------|------------|--------------------|---------------------------|
| 0x21a4 | 7:5 | 0x0 | Function reserved |
| | 4 | 0x0 | reg_PCIE_Refclk_tx_en_byp |
| | 3:0 | 0x0 | Function reserved |
| 0x21a8 | 7:3 | 0x0 | Function reserved |
| | 2 | 0x0 | pcie_device_en |
| | 1 | 0x0 | pcie_refclk_sel |
| | 0 | 0x1 | reg_PCIE_Refclk_tx_enb |
| 0x21ec | 7:2 | 0x2 | Function reserved |
| | 1 | 0x0 | pcie_ckout_sel |
| | 0 | 0x0 | Function reserved |

For more detail information, please refer to section PCIe2/USB3 PHY Detail Register Description for PHY register description.

10.6.1.3 Reset Overview

10.6.1.3.1 Power-On Reset

The power-on reset is used as cold reset of the PCIe. The entire module is reset when power-on reset is asserted. After de-assertion of the power-on reset, the PCIe PHY and PCIe Core function reset keep until the software release them. Application should finish PCIe PHY configuration and wait for TX PLL lock before release reset.

10.6.1.3.2 System reset

The PCIe Controller has the following distinct resets, all of these are configurable through software driver. This section describes the function of each of the reset inputs.

- button_RST_N: Button reset from board if related IO has been used. Please refer to Interface Description section for more information. This reset has an equal functionality with power-on reset.
- perst_N: This reset works as Warm Reset of PCIe if related IO has been used. Please refer to Interface Description section for more information.
- core_RST: This is the main reset for the PCIe Controller. It resets all the logic in the core running in the CORE_CLK domain, except for the PMC module. It keeps reset with power-up reset by default. It also should be asserted when Hot Reset or link down reset occurs, but Application software can delay core_RST_N been asserted until system is ready to do core reset. This delayed reset operation will be discussed in next section.
- AXI Reset: All three AXI interface has its own reset input: mstr_aresetn, slv_aresetn, dbi_resetn. Each resets all logic in the specific AXI_CLK domain. Typically, these resets are be asserted whenever core_RESET is asserted. Application software shall have the knowledge of choosing an appropriate time to drive this reset in order to avoid times when a transaction is in flight.
- non_sticky_RST_N: Resets all non-sticky bit registers in the configuration register space.
- sticky_RST_N : Resets all sticky bit registers in the configuration register space.
- pwr_RST_N: Resets the PMC module and resets all registers in the PM clock domain, including sticky bits.
- phy_RST_N: output reset signal to reset the PCIe PHY. This reset asserted along with Power-On Reset and Hot Reset.

PMC of PCIe Controller may request some system reset such as core_RST_N, sticky_RST_N, non_sticky_RST_N and pwr_RST_N. Application can use PCIE_CLIENT_LOCAL_CRU_CTRL register in client register group to disable these requests. Refer to PCIe Client Detail Register Description for more details.

10.6.1.3.3 Hot Reset and Link-Down Reset

A downstream port (DSP) can hot reset an upstream port (USP) by sending two consecutive TS1 ordered sets with the hot reset bit asserted. Eventually, the DSP and USP assert link_req_rst_not to request external logic to reset them. Alternatively, during normal operation, the link might fail and go down. After this link-down event, the controller requests the reset module to hot-reset the controller. There is no difference in the handling of a link-down reset or a hot reset; the controller core asserts the link_req_rst_not output requesting the reset module to reset the controller.

Hot Reset or Link-down reset cannot be activated before no transaction is appending on system bus. Application software must confirm that system bus is in IDLE state then let the reset operation go on.

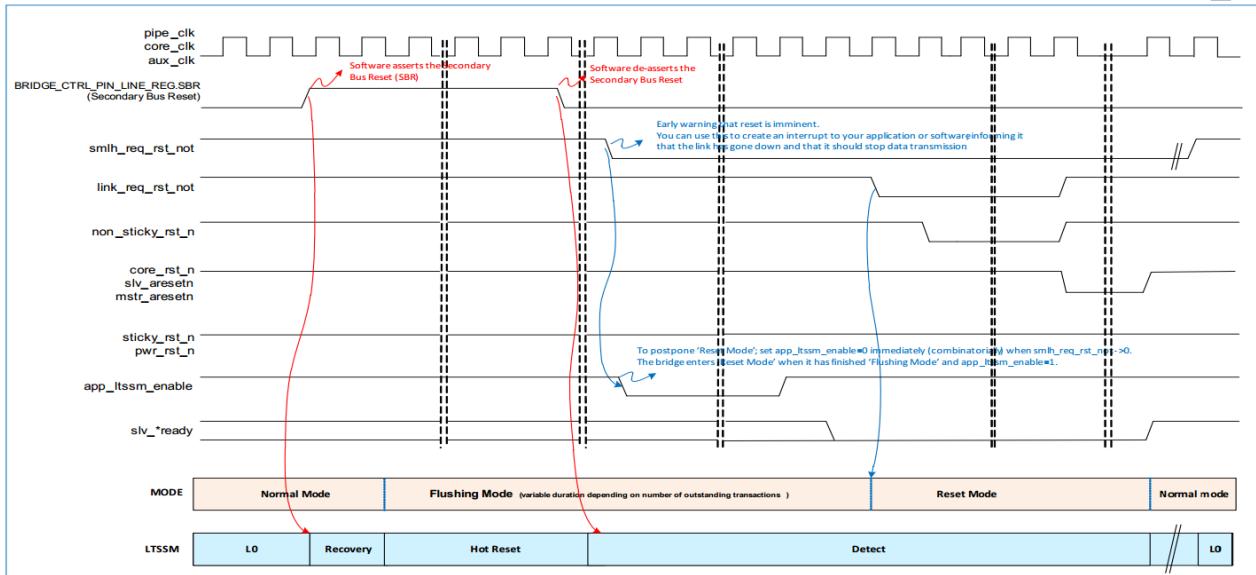


Fig. 10-5 Delaying the PCIe Hot Reset

Figure above illustrate the timing of handling Hot Reset. When Hot Reset or Link-down reset occurs, controller will assert smlh_req_rst_not as an early warning. This warning is an interrupt bit in Client Register group. app_ltssm_enable should be deasserted immediately to disable the LTSSM. Then the Hot Reset process is holding on. Software then should set the PCIe NIU in IDLE state by using PCIE GRF registers. After that, software can assert app_ltssm_enable to let the controller to finish the reset.

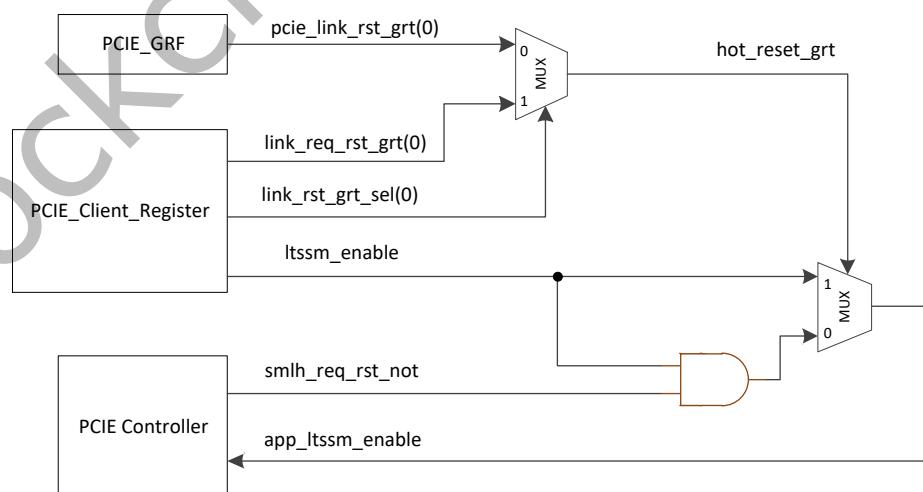


Fig. 10-6 PCIe Hot Reset Control Structure

Figure above illustrate the Hot Reset control structure, when application deassert link_rst_grt, the app_ltssm_enable will be gated if smlh_req_rst_not is asserted. Configure hot_reset_grt by PCIE_GRF register is recommend and the client register control path is reserved for future use.

10.6.2 Initialization

10.6.2.1 Initialization Sequence for RC Mode

The initialization sequence for RC mode is as follows:

1. Configure the GPIO configure for CLKREQ# and WAKE# signals. Configure the optional signals button_rst_n and perst_n if they are used. Please refer to section Interface Description for details.
2. Configure the CRU registers to provide reference clock to PCIe PHY. Please refer to CRU Chapter for more information.
3. Configure USB3PHY General Register File to set PIPE signals which are unused for PCIe to appropriate values.
4. Configure PHY registers to set PCIe PHY work in PCIe RC Mode. Configure Client Registers to set the PCIe Controller device_type to RC mode.
5. Configure PHY registers to set TX PLL parameters to appropriate values. For programming details, please refer to section 10.7.1 Programming setting for PCIe PHY TX PLL.
6. Wait for PCIe PHY TX PLL lock, then release software reset for PCIe Controller.
7. Assert pcie_link_rst_grt in PCIE_GRF (refer to section Hot Reset and Link-Down Reset) to free ltssm_enable to prepare for link training. This step must be down because smlh_req_rst_not is asserted after power-on reset until first link up.
8. You can program controller registers before Link training start if necessary. Link training can be initialed by setting ltssm_enable bit in register PCIE_CLIENT_GENERAL_CON to 1. Link training success will set phy_link_up_int interrupt status bit in register PCIE_CLIENT_INTR_STATUS_MISC to 1. And if data flow initialization success, dli_link_up_int interrupt status bit will be set to one in same register. To Initially, the controller tries to linkup at maximum Link width, When you have unused lanes in your system, you must tie them off using the hardware and software procedures outlined in section Tie off Unused Lanes.
Link speed will change to Gen2 automatically if you set direct_speed_change bit to 1 in PL_GEN2_CTRL_OFF register and both device support Gen2 speed.
9. RC finish the device Enumeration. For example: read the configuration space of the downstream devices, program device capabilities, program the BARs of endpoints.
10. Writes to Bus Master Enable (BME), Memory Space Enable (MSE), then application can start application traffic generation.

10.6.2.2 Initialization Sequence for EP Mode

The initialization sequence for RC mode is as follows:

1. Configure the GPIO configure for CLKREQ# and WAKE# signals. Configure the optional signals button_rst_n and perst_n if they are used. Please refer to section Interface Description for details.
2. Configure the CRU registers to provide reference clock to PCIe PHY. Please refer to CRU Chapter for more information. Configure PCIe PHY register if external differential clock is use as reference clock.
3. Configure USB3PHY General Register File to set PIPE signals which are unused for PCIe to appropriate values.
4. Configure PHY registers to set PCIe PHY work in PCIe EP Mode. Configure Client Registers to set the PCIe Controller device_type to EP mode.
5. Configure PHY registers to set TX PLL parameters to appropriate values. For programming details, please refer to section 10.7.1 Programming setting for PCIe PHY TX PLL.
6. Wait for PCIe PHY TX PLL lock, then release software reset for PCIe Controller.
7. Assert pcie_link_rst_grt in PCIE_GRF (refer to section Hot Reset and Link-Down Reset) to free ltssm_enable to prepare for link training. This step must be down because smlh_req_rst_not is asserted after power-on reset until first link up.
8. You can program controller registers before Link training start if necessary. Link training can be initialed by setting ltssm_enable bit in register PCIE_CLIENT_GENERAL_CON to 1. Link training success will set phy_link_up_int interrupt status bit in register PCIE_CLIENT_INTR_STATUS_MISC to 1. And if data flow initialization success,

- `dll_link_up_int` interrupt status bit will be set to one in same register. To Initially, the controller tries to linkup at maximum Link width, When you have unused lanes in your system, you must tie them off using the hardware and software procedures outlined in section Tie off Unused Lanes.
- Link speed will change to Gen2 automatically if you set `direct_speed_change` bit to 1 in `PL_GEN2_CTRL_OFF` register and both device support Gen2 speed.
9. The PCIe then can accept configuration setup access, after emulation done, the transactions can be initiated.

10.6.2.3 Tie Off Unused Lanes

When the link width of the PHY is smaller than the link width of the controller, you must tie off the unused lanes of the controller's PIPE interface.

Following PHY input signal pins should be assigned proper values by configuring PCIe PHY GRF:

```
'pipe_pd_i_I0/1[1:0]'=2'b11
'pipe_txelecidle_i_I0/1'=1'b1
'pipe_txdetectrx_lb_i_I0/1'=1'b0
'pipe_txdata_i_I0/1[31:0]'=32'h0
'pipe_txdatak_i_I0/1[3:0]'=4'h0
'pipe_ebuffmode_i_I0/1'=1'b0
'pipe_rxterm_i_I0/1'=1'b0
'pipe_txcompliance_i_I0/1'=1'b0
'pipe_txoneszeros_i_I0/1'=1'b0
```

Following registers should be reprogram to proper values:

- `LINK_CAPABLE` field of the `PORT_LINK_CTRL_OFF` register to 6h1 from 6h7. This is used by the LTSSM in Detect.
- `NUM_OF_LANES[8:0]` field of the `GEN2_CTRL_OFF` register. This indicates to the LTSSM, the number of lanes to check for exiting from L2.Idle or Polling.Active.
- `PCIE_CAP_MAX_LINK_WIDTH` field of the `LINK_CAPABILITIES_REG` register

10.6.3 Address Translation

The controller uses the internal Address Translation Unit(iATU) to implement a local address translation scheme that replaces the TLP address and TLP header fields in the current TLP request header.

The iAUT has 16 inbound Address Translation regions and 16 outbound Address Translation regions. The minimum size of an Address Translation Region is 64k and the maximum size of an Address Translation Region is 4G.

10.6.3.1 Outbound Features

Address translation is used for mapping different address ranges to different memory spaces supported by your application. A typical example maps your application memory space to PCI memory space. The ATU also supports type translation. Without address translation, your application address is passed unmodified to the TLPs directly through the Tx application interface. You can program the iATU to implement your own outbound address translation scheme. The outbound features is as follows:

- Address Match mode operation for MEM and I/O, CFG, and MSG TLPs. No translation for completions.
- Supports type translation through TLP type header field replacement for MEM or I/O types to MSG/CFG types.
- Programmable TLP header field replacement. Including TYPE, TC, AT, ATTR, MSG-Code, TH, PH, ST.
- Multiple (up to 16) address regions programmable for location and size.
- Programmable enable/disable per region.
- Automatic FMT field translation between three DWORDs and four DWORDs for 64-bit addresses.
- Invert Address Matching mode to translate accesses outside of a successful address match.

- Configuration Shift mode. Optimizes the memory footprint of CFG accesses destined for the Rx application interface in a multifunction device.
- Response code which defines the completion status to return for accesses matching a region.
- Supports regions from 64 KB to 4 GB in size.
- Payload Inhibit marks all TLPs as having no payload data.
- Header Substitution replaces bytes 8 to 11 (for 3 DWORD header) or bytes 12 to 15 (for 4 DWORD header), inclusive, of the outbound TLP header.
- Tag Substitution of the outbound TLP tag field.
- Function number bypass mode to allow function number information to be supplied from your application transmit interface while translating the address and other attributes of the TLP.
- DMA bypass mode to allow TLPs which are initiated by the embedded DMA engine, to pass through the iATU untranslated.

10.6.3.2 Outbound Basic Operation

The address field of each request MEM and I/O TLP is checked to see if it falls into any of the enabled address regions defined by the Start and End addresses as defined in Figure below. When an address match is found, then the TLP address field is modified as follows:

$$\text{Translated Address} = \text{Original Address} - \text{Base Address} + \text{Target Address}$$

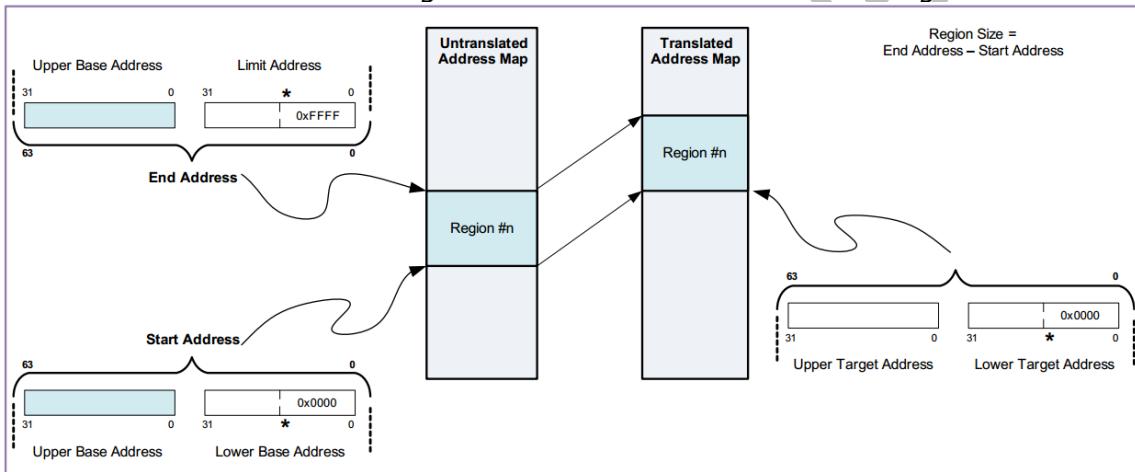


Fig. 10-7 PCIe Hot Reset Control Structure

The TYPE, TC, AT, TH, PH, ST, Function Number, and ATTR header fields are replaced with the corresponding fields in the IATU_REGION_CTRL_1_OFF_OUTBOUND_0 register. When your application address field matches more than one of the CX_ATU_NUM_OUTBOUND_REGIONS address regions, then the first (from lowest number 0) enabled region to be matched is used. For details on what happens when there is no address match, see "No Address Match Result". This operational mode (called Address Match Mode) is always used for outbound translation.

The minimum size of an address translation region is 64KB. So the lower 16 bits of the Base, Limit, and Target registers are zero and all address regions are aligned on 64 KB boundaries.

10.6.3.3 Outbound Detailed Operation

10.6.3.3.1 RID BDF Number Replacement

When there is a successful address match on an outbound TLP, then the function number used in generating the function part of the requester ID field of the TLP is taken from the 3-bit Function Number field of the IATU_REGION_CTRL_1_OFF_OUTBOUND_register. The value in this field must be 0x0 unless multifunction operation in the controller is enabled. To override this behavior, use the "Function Number Translation Bypass Feature" described later.

10.6.3.3.2 iATU Outbound MSG Handling

The iATU supports TYPE translation/conversion of MEM and I/O TLPs to Msg/MsgD TLPs. This supports applications that are unable to directly generate Msg/MsgD TLPs. When there is a successful address match on an outbound MEM TLP, and the translated TLP type field is MSG (that is, the type field of the IATU_REGION_CTRL_1_OFF_OUTBOUND_i register is 10xxx), then the message code field of the TLP is set to the value in the Message Code field of the IATU_REGION_CTRL_2_OFF_OUTBOUND_i register. A Memory Write with an effective length of '0' is converted to Msg and all other MWr TLPs are converted to MsgD.

10.6.3.3.3 MEM-CFG Type Translation

The iATU supports translation of I/O and MEM TLPs to CFG TLPs. This is useful for applications that are unable to generate CFG TLPs. The 16-bit BDF is located at bits [31:16] of the translated address where:

$$\text{Translated Address} = \text{Original Address} - \text{Base Address} + \text{Target Address}$$

As an example:

$$\text{Original Address}[31:16] = \{13h0, function_no[2:0]\}$$

$$\text{Base Address}[31:16] = 16h0$$

$$\text{Target Address}[31:16] = \{\text{bus_no}[7:0], \text{device_no}[4:0], 3h0\}$$

then:

$$\text{Translated Address}[31:16] = \text{BDF} = \{\text{bus_no}[7:0], \text{device_no}[4:0], \text{function_no}[2:0]\}$$

To handle eight functions (as the previous example indicates), you should use a 19-bit wide region size. For CFG transactions created directly by your application (as opposed to the iATU), you must ensure that the BDF field does not match any programmed iATU address region or else unintentional type translation could occur.

10.6.3.3.4 CFG Shift Feature

This feature is enabled by setting the CFG_SHIFT_MODE field the IATU_REGION_CTRL_2_OFF_OUTBOUND_0 register. The iATU uses bits [27:12] of the original address to form bits [31:16] (BDF location) of the outgoing CFG TLP. This supports the Enhanced Configuration Address Mapping (ECAM) mechanism by allowing all outgoing I/O and MEM TLPs (that have been translated to CFG) to be mapped from memory space into any 256 MB region of the PCIe configuration space.

10.6.3.3.5 FMT Translation

The iATU automatically sets the TLP format field for three DWORDs when it detects all zeros in the upper 32 bits of the translated address. Otherwise, it sets it to four DWORDs when it detects a 64-bit address (that is, when there is a '1' in the upper 32 bits of the translated address). When the original address and the translated address are of different format, the iATU ensures that the TLP header size matches the translated address format.

10.6.3.3.6 Invert Feature

In normal operation an address match on an outbound TLP occurs when the untranslated address is in the region bounded by the base address and limit address. When the invert feature is activated, an address match occurs when the untranslated address is not in the region bounded by the base address and limit address. This feature is activated by setting the Invert field of the IATU_REGION_CTRL_2_OFF_OUTBOUND_i register.

10.6.3.3.7 DMA Bypass Feature

When you do not want the iATU to translate outbound requests that are generated by the DMA, you must implement one of the following approaches:

- Ensure that the combination of DMA channel address programming and iATU control register programming causes no translation of DMA traffic to be done in the iATU.
- Activate the DMA bypass mode to allow request TLPs which are initiated by the embedded DMA controller to pass through the iATU untranslated. You can activate the DMA bypass mode by setting the DMA Bypass field of the IATU_REGION_CTRL_2_OFF_OUTBOUND_i register to 1.

10.6.3.3.8 Function Number Translation Bypass Feature

In this mode the function number of the translated TLP is taken from your application transmit interface and not from the Function Number field of the REGION_CTRL_1_OFF_OUTBOUND_i register. You can activate the function number bypass mode by setting the Function Number Translation Bypass Enable field in the IATU_REGION_CTRL_2_OFF_OUTBOUND_i to '1'.

10.6.3.3.9 General Bypass

Application can program PCIE_CLIENT_AXI_SLV_ATU_BYPASS register to do general ATU bypassing. Note that you should make sure that there is no transfer pending before you program this register.

10.6.3.3.10 Header Substitution (Tx)

When enabled and region address is matched, the iATU fully substitutes bytes 8-11 (for 3 DWORD header) or bytes 12-15 (for 4 DWORD header) of the outbound TLP header with the contents of the LWR_TARGET_RW field in IATU_LWR_TARGET_ADDR_OFF_OUTBOUND_i. The expected usage scenario is for Vendor Defined Msg/MsgD and ATS transactions over the AXI bridge which is normally inefficient requiring a very large iATU region. Enabled using the HEADER_SUBSTITUTE_EN field in IATU_REGION_CTRL_2_OFF_OUTBOUND_i.

10.6.3.3.11 Tag Substitution (Tx)

When enabled and region address is matched, the iATU substitutes the TAGfield of the outbound TLP header with the contents of the TAG field in IATU_REGION_CTRL_2_OFF_OUTBOUND_i. The expected usage scenario is translation from AXI MWr to Vendor Defined Msg/MsgD. Enabled using the TAG_SUBSTITUTE_EN field in IATU_REGION_CTRL_2_OFF_OUTBOUND_i.

Your application must not attempt to perform TAG substitution for outgoing non-posted TLPs.

Note: If the iATU is programmed to allow a TLP to be matched to more than one iATU outbound regions, the Function Number Translation Bypass field of the IATU_REGION_CTRL_2_OFF_OUTBOUND_i register should be same for every region where that TLP can be matched.

10.6.3.3.12 No Address Match Result

When there is no address match then the address is untranslated but the TLP header information (for fields that are programmable) comes from the relevant fields on the application transmit interface AXI slave.

10.6.3.4 Outbound Programming Example

Define Outbound Region 1 as:

64 KB I/O region from 0x80000000_d0000000 to 0x80000000_d000ffff, to be mapped to 0x00010000 in the PCIe I/O space.

1. Setup the Region Base and Limit Address Registers.

 Write 0xd0000000 to Address {0x208} to set the Lower Base Address.

 Write 0x80000000 to Address {0x20C} to set the Upper Base Address.

 Write 0xd000ffff to Address {0x210} to set the Limit Address.

2. Setup the Target Address Registers.

 Write 0x00010000 to Address {0x214} to set the Lower Target Address.

 Write 0x00000000 to Address {0x218} to set the Upper Target Address.

3. Configure the region through the Region Control 1 Register.

 Write 0x00000002 to Address {0x200} to define the type of the region to be I/O.

4. Enable the region.

 Write 0x80000000 to Address {0x204} to enable the region.

10.6.3.5 Inbound Features

Address translation is used for mapping different address ranges to different memory spaces supported by your application. A typical example maps your application memory space to PCI memory space. The iATU supports type translation. Without address translation, your application address is passed from the TLPs directly through the AXI application interface. You can program the iATU to implement your own inbound address translation scheme without external logic.

- Programmable Match mode operation for MEM, I/O, CFG, and MSG TLPs. No translation for completions.
- Selectable BAR Match mode operation for I/O and MEM TLPs:
TLPs destined for the internal CDM (or ELBI) in an upstream port are not translated.
TLPs that are not error-free (ECRC, malformed and so on) are not translated.
- Programmable TLP header field matching:
TYPE/TD/TC/AT/ATTR/MSG-Code/TH/PH/ST
 - Function Number
- Multiple (up to 16) address regions programmable for location and size.
- Programmable enable/disable per region.
- Automatic FMT field translation between three DWORDs and four DWORDs for 64-bit addresses.
- Invert Address Matching mode to translate accesses outside of a successful address match.
- ECAM Configuration Shift mode to allow a 256 MB CFG1 space to be located anywhere in the 64-bit address space.
- Supports regions from 64 KB to 4 GB in size.
- Single Address Location to allow all TLPs to be translated to a single address location.
- Msg Type Match Mode to allow matching of any TLP of type Message.

10.6.3.6 Inbound Basic Operation

10.6.3.6.1 Overview

The following translation rules and limitations apply:

- When there is no match, then the address is untranslated. In addition
- TLPs destined for the internal registers in an upstream port are not translated.
- TLPs that are not error-free (ECRC, malformed and so on) are not translated.
- Address translation of all TLP types (MEM, I/O, CFG, and MSG) except completion is supported in Address Match mode. In BAR Match mode, only translation of I/O and MEM is supported.

The setting of the MATCH_MODE field in IATU_REGION_CTRL_2_OFF_INBOUND_0 determines how iATU inbound matching is done for each TLP type.

Table 10-6 Determination of Match Mode

| TLP Type | MATCH_MODE = 0 | MATCH_MODE = 1 |
|------------|-----------------------|----------------------|
| MEM or I/O | Address Match Mode | BAR Match Mode |
| CFG0 | Routing ID Match Mode | Accept Mode |
| MSG/MSGD | Address Match Mode | Vendor ID Match Mode |

10.6.3.6.2 I/O and MEM Match Modes

Inbound address translation for I/O and MEM TLPs operates in one of two matching modes as determined by the "Inbound Match Mode" field in the IATU_REGION_CTRL_2_OFF_INBOUND_0 register.

Address Match Mode: The operation is similar to "Outbound Basic Operation (Address Match Mode)". The address field of each request TLP is checked to see if it falls into any of the enabled address regions. When an address match is found then the TLP address field is modified as follows:

$$\text{Address} = \text{Address} - \text{Base Address} + \text{Target Address}$$

BAR Match Mode:

Looking for an address match is a two-step process.

1. The standard internal PCI Express BAR Matching Mechanism checks if the address field

- of any MEM and I/O request TLP falls into any address region defined by the enabled BAR addresses and masks.
- When a matched BAR is found, then the iATU compares the BAR ID to the BAR Number field in the IATU_REGION_CTRL_2_OFF_INBOUND_0 register for all enabled regions. Figure below provides more details on inbound translation in BAR Match Mode.

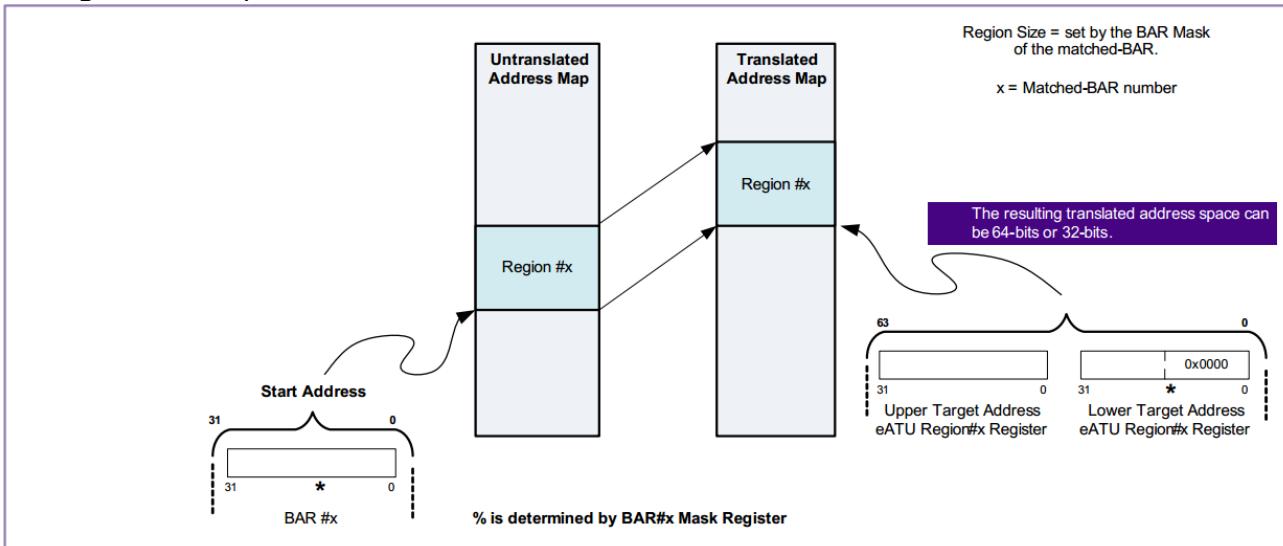


Fig. 10-8 Inbound BAR Match Mode

10.6.3.7 Inbound Detailed Operation

10.6.3.7.1 Single Address Location

When enabled and region address is matched, the TLPs can be translated to a single address location as determined by the target address register of the iATU region. This feature can be enabled using the SINGLE_ADDR_LOC_TRANS_EN field in IATU_REGION_CTRL_2_OFF_INBOUND_i. This feature is useful for translating messages on AXI bridge.

10.6.3.7.2 CFG Handling (Upstream Port)

The controller normally routes CFG TLPs (to the internal CDM or ELBI) without translation. The iATU only translates CFG0 TLPs that the controller has routed to the TRGT1. Inbound address translation for CFG0 TLPs operates in one of two matching modes as determined by the Inbound CFG0 Match Mode field in the IATU_REGION_CTRL_2_OFF_INBOUND_i register.

Accept Mode

The controller always accepts CFG0 TLPs even when the CFG bus number does not match the current bus number of the device. This mode follows that behavior. The routing ID of received CFG0 TLPs are ignored when determining a match.

Routing ID Match Mode

The operation is similar to "Outbound Basic Operation (Address Match Mode)". The routing ID of the inbound CFG0 TLP must fall within the Base and Limit of the defined iATU region for matching to proceed. The iATU interprets the routing ID (Bytes 1 8-11 of TLP header) as an address. This corresponds to the upper 16 bits of the address in MEM and I/O transactions.

CFG Shift Feature

Inbound CFG transactions routed to the Rx application interface can exist anywhere in address space, because the PCIe controller filter processes the routing ID (BDF) as bits [31:16] of an address. This BDF changes according on the PCIe bus topology. A compressor feature (CFG Shift Feature) can be enabled by setting the CFG Shift bit in the IATU_REGION_CTRL_2_OFF_INBOUND_i register. Bits [15:12] of the third DWORD1 of CFG TLPs are reserved. The compressor feature uses this fact to reduce the memory requirement. This shifts/maps the BDF (bits [31:16] of the third header DWORD, which would be matched against the Base and Limit addresses) of the incoming CfgRd0/CfgWr0

down to bits [27:12] of the translated address.

10.6.3.7.3 Optional Matching Fields

In address and BAR match modes, a successful address/BAR match can be optionally gated by successful matching of the following programmable TLP header fields (per region):

- TYPE/TD/TC/AT/ATTR/TH/PH/ST
- MSG Code (MSG TLPs only)
- Function number (MEM, I/O, or CFG TLPs only)
- Virtual function number (MEM or I/O TLPs only)

For each of the previous fields in the IATU_REGION_CTRL_1_OFF_INBOUND_i register, there is an associated Match Enable bit in the IATU_REGION_CTRL_2_OFF_INBOUND_i register. Address translation only proceeds when compares on all enabled field are successful.

10.6.3.7.4 Response Code Feature

When the Response Code field of the IATU_REGION_CTRL_REG_2_INBOUND_i register is set to a value other than 00, the controller uses it to determine the completion status field of completion TLPs sent in response to successfully matched non-posted TLPs. This can be set to unsupported request (UR) or completer abort (CA). When the error response field is set to 2b00, then the normal receive filter response for this TLP is used.

10.6.3.7.5 Inbound MSG Handling

Inbound message (Msg/MsgD) transactions can use one of two matching modes:

Address Match Mode

The third and fourth header DWORDs are treated as an address and are compared against the iATU Region Base and Limit Address registers. For vendor defined messages this allows specific messages to be filtered into memory at the target address. The Upper Base address should be set to BDF and Vendor ID. The Lower Base address can be used as a filter for specific messages.

Vendor ID Match Mode

This mode is relevant for ID-routed vendor defined messages. The iATU ignores the routing ID (BDF) in bits [31:16] of the third DWORD of the TLP header1, but compares it against the vendor ID in bits [15:0] of the third DWORD of the TLP header (bytes1 10 and 11). This allows vendor defined messages to be filtered against specific vendor IDs without needing to know the BDF number which might vary depending on the PCI topology. Bits [15:0] of the Region Upper Base register should be programmed with the required vendor ID as follows:

Region Upper Base [15:8] =byte 10

Region Upper Base [7:0] =byte 11

The lower base and limit register should be programmed to translate TLPs based on vendor-specific information in the fourth DWORD of the TLP header.

10.6.3.7.6 Msg Type Match Mode

Inbound message (Msg/MsgD) transactions can also use Msg Type Matching mode. When this mode is enabled and Single Address Location is enabled, the iATU matches Msg TLP Type field with TYPE field of IATU_REGION_CTRL_1_OFF_INBOUND_i register.

If Fuzzy Type Match Mode is also enabled, then any Msg received will be matched (that is, Msg Type's sub-field r[2:0], which specifies the Message routing mechanism, is ignored). The Message should be consumed by your application before the next message arrives as all messages go to the same address.

If SINGLE_ADDRESS_LOCATION_TRANSLATE_EN is set for any region, then you must ensure that the same TLP cannot be matched in any other region where SINGLE_ADDRESS_LOCATION_TRANSLATE_EN is not set. If this happens raddr_trgt1_hdr_uppr_bytes could have incorrect data.

10.6.3.7.7 Fuzzy Type Match Mode

When enabled, the iATU relaxes the matching of the TLP type field against the expected type field so that:

- CfgRd0 and CfgRd1 TLPs are seen as identical. Similar with CfgWr0 and CfgWr1.
- MWr, MRd and MRdLk TLPs are seen as identical.
- The routing field of MsgD TLPs is ignored.
- Atomic Ops TLPs-FetchAdd, Swap, and CAS are seen as identical.

For example, CFG0 in the type field in the IATU_REGION_CTRL_1_OFF_INBOUND_i register matches against an inbound CfgRd0, CfgRd1, CfgWr0, or CfgWr1 TLP. To enable this feature, set the Fuzzy Type Match Mode bit of the IATU_REGION_CTRL_OFF_2_INBOUND_i register.

10.6.3.7.8 FMT Translation

The iATU automatically sets the TLP format field for three DWORDs when it detects all zeroed in the upper 32 bits of the translated address. Otherwise it sets it to four DWORDs when it detects a 64-bit address (when there is a 1 in the upper 32 bits of the translated address). When the original address and the translated address are of a different format then the iATU ensures that the TLP header size matches the translated address format.

10.6.3.7.9 Invert Feature

Normally an address match on an inbound TLP occurs when the untranslated address is in the region bounded by the Base address and Limit address. When the Invert feature is activated, an address match occurs when the untranslated address is not in the region bounded by the Base address and Limit address. This feature is activated by setting the Invert field of the IATU_REGION_CTRL_OFF_2_INBOUND_i register.

10.6.3.8 Inbound Programming Example

You must not update the iATU registers while operations are in progress on the AXI bridge slave interface.

Example 1:

Define Inbound Region 2 as: MEM region matching BAR4 (BAR match mode) mapping to 0x8000_0000_2000_0000 in your application memory space

1. Setup the Target Address Registers.

 Write 0x20000000 to Address {0x508} to set the Lower Target Address.

 Write 0x80000000 to Address {0x50C} to set the Upper Target Address.

2. Configure the region through the Region Control 1 Register.

 Write 0x00000000 to Address {0x500} to define the type of the region to be MEM.

3. Enable the region for BAR Match Mode.

 Write 0xC0000400 to Address {0x504} to enable the region for BAR match mode for BAR#4.

Example 2:

MEM region matching TLPs with addresses in the range 0x00010000 to 0x0005ffff mapped to 0x2000_0000 - 0x2004_ffff in your application memory space.

1. Setup the Region Base and Limit Address Registers.

 Write 0x00010000 to Address {0x108} to set the Lower Base Address.

 Write 0x00000000 to Address {0x10C} to set the Upper Base Address.

 Write 0x0005ffff to Address {0x110} to set the Limit Address

2. Setup the Target Address Registers.

 Write 0x20000000 to Address {0x114} to set the Lower Target Address.

 Write 0x10000000 to Address {0x118} to set the Upper Target Address.

3. Configure the region through the Region Control 1 Register.

 Write 0x00000000 to Address {0x100} to define the type of the region to be MEM.

4. Enable the region.

 Write 0x80000000 to Address {0x104} to enable the region in address match mode

10.6.4 PCIe Embedded DMA

10.6.4.1 PCIe DMA Overview

The RC system CPU, or the EP application CPU, can off load the transfer ring of large blocks of data to the embedded DMA controller, leaving the CPU free to perform other tasks. You can configure the DMA to have one to eight read channels and one to eight write channels. It can simultaneously perform the following types of memory transactions:

DMA write

Transfer (copy) of a block of data from local memory to remote memory.

DMA read

Transfer (copy) of a block of data from remote memory to local memory.

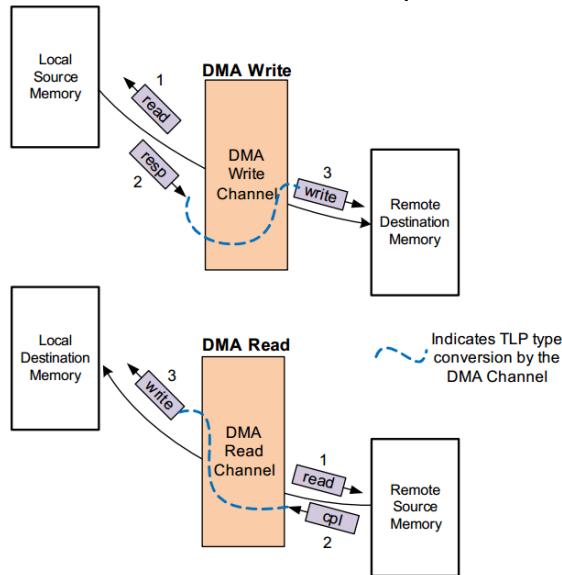


Fig. 10-9 System Level View of PCIe Embedded DMA

Therefore the DMA supports full duplex operation, processing read and write transfers at the same time, and in parallel with normal (non-DMA) traffic. Upon completion of a DMA transfer or an error, the DMA optionally interrupts the local CPU or sends an interrupt MW_r (IMWr) to the remote CPU. The DMA is highly configurable and you can program it using the local DBI or over the PCIe wire.

In linked list mode, the DMA fetches the transfer control information (called channel context) for each transfer (block), from a list of DMA elements that have constructed in local memory.

10.6.4.2 Interrupts and Error Handling

The DMA generates two interrupts, read channel interrupt and write channel interrupt. Each of them can be caused by one of the two reasons:

Done: The DMA successfully completes the transfer.

Abort: The DMA fails to complete the transfer, or an error occurs during the transfer.

The interrupts are signaled to the software on your CPU, using one or both of the following mechanisms:

- Locally through the edma_rd_int or edma_wr_int field of PCIE_CLIENT_INTR_STATUS_MISC register.
- Remotely using a posted memory write (IMWr), which can be interpreted as an MSI or MSIX when directed toward the RC.

For remote interrupt, there are two programmable IMWr addresses per channel, one each for the done and abort interrupts. For MSI, you must program all IMWRr address registers with the same MSI address, as PCIe only supports a single MSI address per function.

A single IMWr data register is used for both types of interrupts, so you must read DMA_READ_INT_STATUS_OFF(or DMA_WRITE_INT_STATUS_OFF) to identify the interrupt type.

The interrupt handling mechanism is different for linked list (LL) mode (than non LL mode), and there are also some differences between the read and write channels.

10.6.4.2.1 Non Linked List Mode Interrupt Handling

You enable the local and remote interrupts through the local and remote interrupt enable (LIE and RIE) bits: DMA_CH_CONTROL1_OFF_WRCH_0.lie and DMA_CH_CONTROL1_OFF_WRCH_0.rie.

In the write channel, there is only one error condition that results in an abort interrupt. For more details, see "Linked List Mode" on next section. You mask, clear, and read the status of each of the two interrupts (done and abort) through the DMA interrupt registers as indicated in figure below:

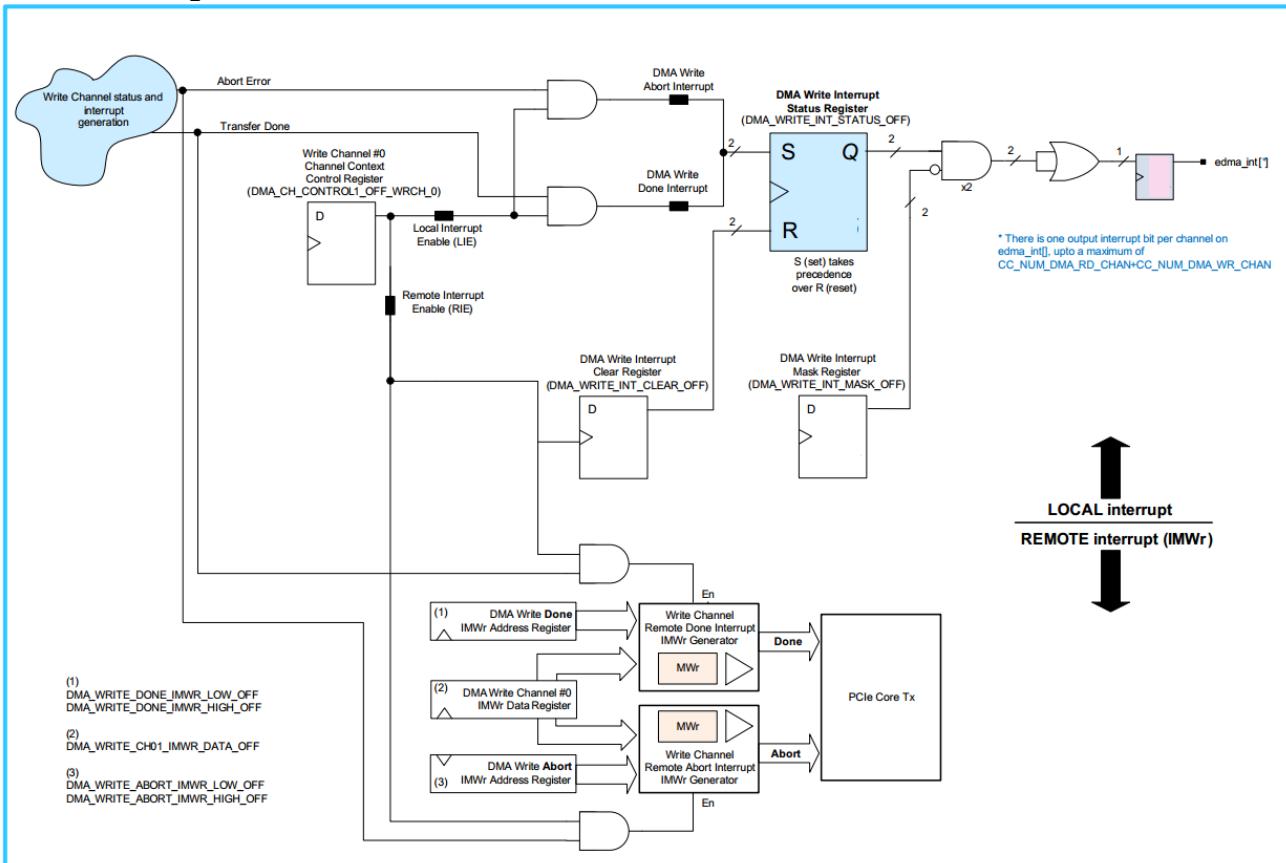


Fig. 10-10 Write Interrupt Generation - Non Linked List Mode

In the read channel, there are five error conditions that results in an abort interrupt. For more details, see "Linked List Mode" on next section. You mask and clear each of the two interrupts (done and abort) through the DMA interrupt registers as indicated in figure below. However, you can read the status of each of the five abort errors (that contribute to the abort interrupt) through DMA_READ_ERR_STATUS_LOW_OFF and DMA_READ_ERR_STATUS_HIGH_OFF.

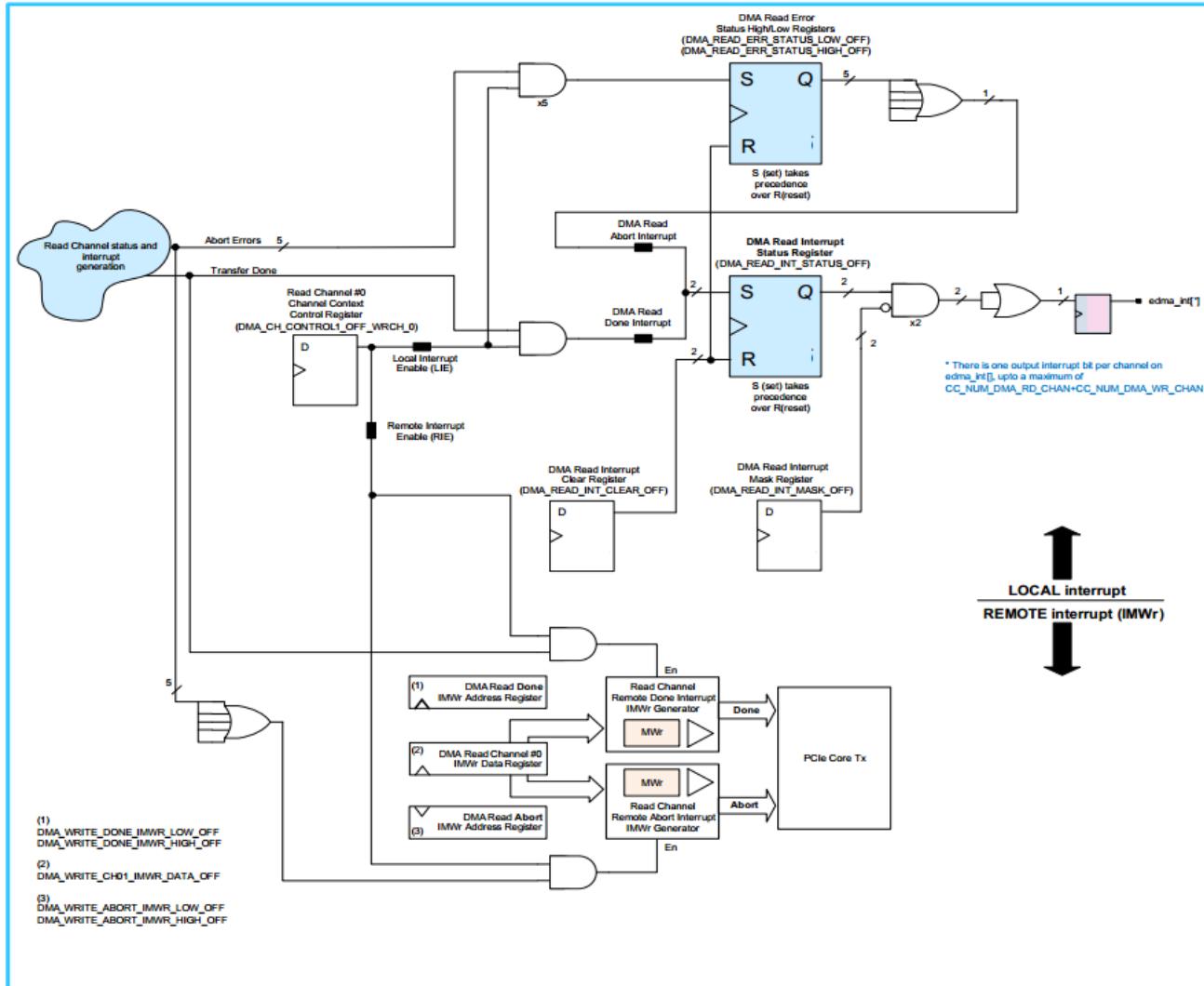


Fig. 10-11 Read Interrupt Generation - Non Linked List Mode

10.6.4.2.2 Linked List Mode Interrupt Handling

The LIE and RIE bits in the LL element enable the channel done interrupts (local and remote). The LLLAIE and LLRAIE bits of the DMA_WRITE_LINKED_LIST_ERR_EN_OFF and DMA_READ_LINKED_LIST_ERR_EN_OFF registers enable the channel abort interrupts (local and remote). In the write channel, there are two error conditions that results in an abort interrupt. For more details, see “Linked List Mode” on next section. You mask and clear each of the two interrupts (done and abort) through the DMA interrupt registers as indicated in figure below. You can read the status of each of the two abort errors (that contribute to the abort interrupt) through the DMA_WRITE_ERR_STATUS register.

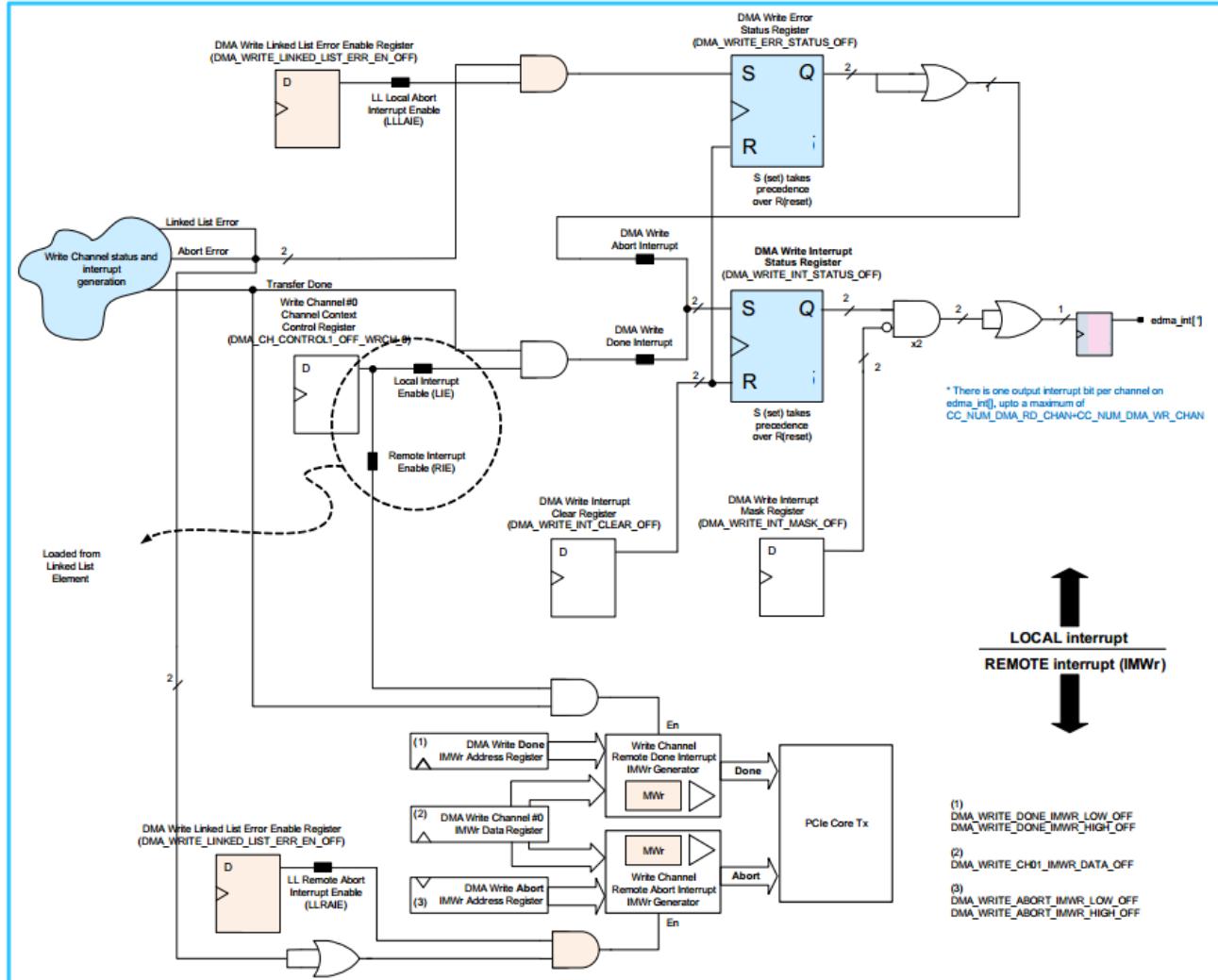


Fig. 10-12 Write Interrupt Generation - Linked List Mode

In the read channel, there are six error conditions that results in an abort interrupt. For more details, see "Linked List Mode" on next section. You mask and clear each of the two interrupts (done and abort) through the DMA interrupt registers as indicated in figure below. You can read the status of each of the six abort errors (that contribute to the abort interrupt) through the DMA_READ_ERR_STATUS_LOW_OFF and DMA_READ_ERR_STATUS_HIGH_OFF registers.

In non-linked list mode, LIE acts as a global switch. However when in linked list mode, LIE is just local to the current linked list element and the global switch is LLLAIE.

If the DMA driver is running on the host and the interrupt service routine is reading local interrupts to determine if the transfer is successful, then you must set LIE and RIE in the same element and you should mask or ignore the local interrupt pin. Setting RIE and LIE in element A followed by RIE (only) in element B is not a verified usage scenario.

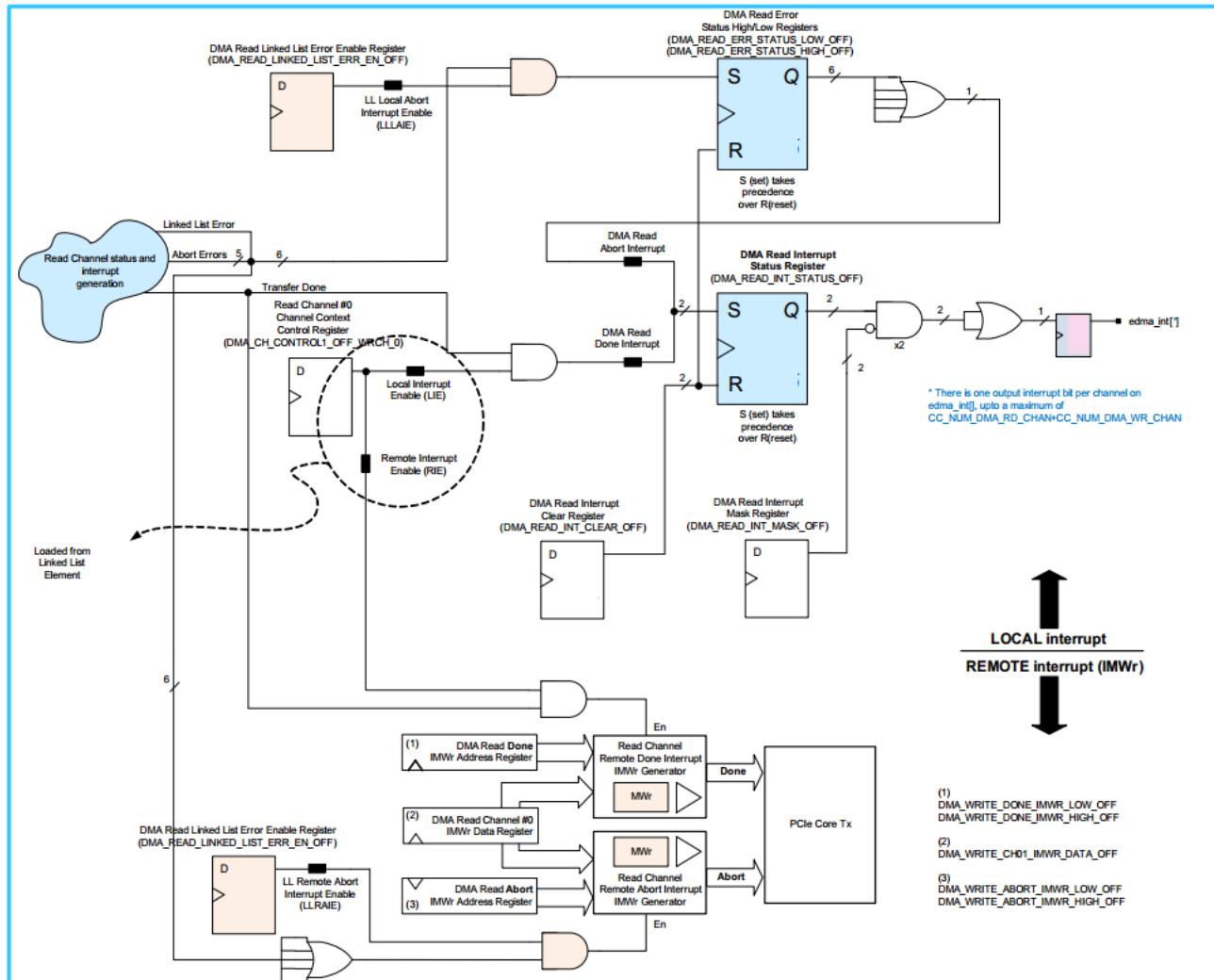


Fig. 10-13 Read Interrupt Generation - Linked List Mode

10.6.4.3 Linked List Mode

The DMA provides a linked list (LL) mode to efficiently move data from source to destination with minimal intervention from the local CPU. This mode provides an alternative to programming the DMA multiple times to transfer multiple blocks of data. The programming information (address, size, and so on) for each block of memory is pre-programmed by your software into a LL element (also known as a descriptor) in local memory. Each element (called a data element) in the LL structure (called a transfer list) can transfer up to 4 GB of data.

You enable LL operation for a channel, by setting the LLE field of the DMA_CH_CONTROL1_OF[WRCH|RDCH]_0 register to 1. Your application must produce the LL element structure in local memory as shown in figure below. Normally, all of the elements are contiguous (one after the other) in memory, and each element has six DWORDs containing the information about the block of data to be transferred. You program the channel context registers (DMA_LL_P_LOW_OF_WRCH_0 and DMA_LL_P_HIGH_OF_WRCH_0) with the location of where you have placed the LL element structure in local memory.

When you start the DMA transfer (by writing to the DMA Write Doorbell Register DMA_WRITE_DOORBELL_OF or DMA Read Doorbell Register DMA_READ_DOORBELL_OF), the DMA reads (consumes) each element from local memory, and loads the information (SAR, DAR, size, and so on) from that element into the channel context registers in the DMA. These channel context registers determine the operation of the channel that the DMA controller is currently servicing. The DMA then proceeds to transfer the block of data (as defined by the element), and when it is finished, reads the next element from local memory. Normally, all of the elements are contiguous (one after the other) in memory, with

the starting address defined in the channel context DMA Linked List Pointer Low Register DMA_LLPO_LOW_OFF_WRCH_0.

When you want to jump in local memory to another element list (or recycle the consumed elements), then you set the LLP bit in the element (for example, link element #N-1 in figure below), specify the location of the next element structure using the LL Element Pointer DWORDs, and, set TCB to 1 (for recycling) or to 0 (to jump to another list).

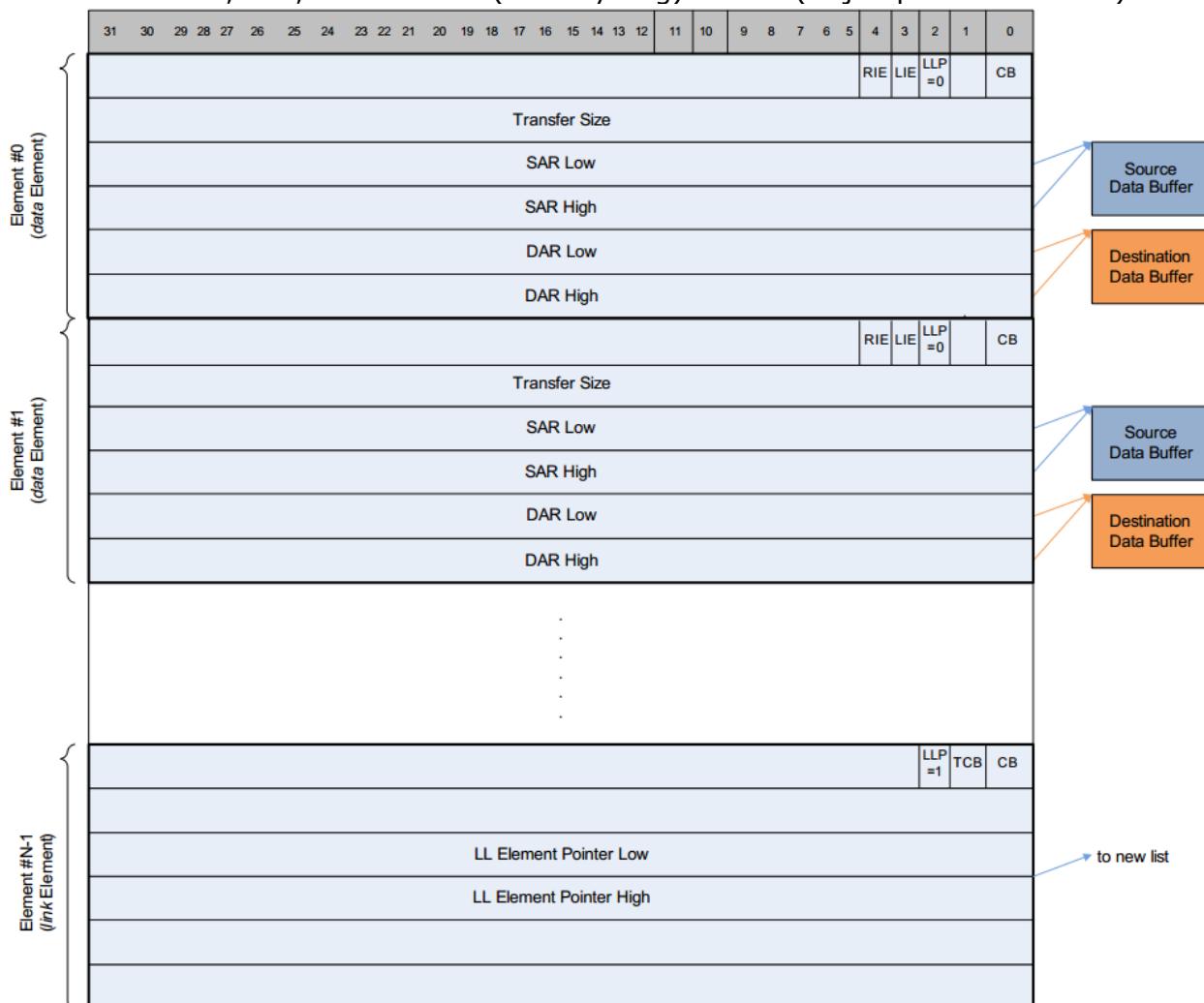


Fig. 10-14 Linked List Element/Descriptor Structure in Local Memory

10.6.4.3.1 LL Element and Channel Context Registers

Notice the similarity between a data element and the DMA Channel Context registers for each channel. Each element has six DWORDs as in Figure 9-6. There are eight channel context registers (DWORDs) for a channel. The DMA loads the six element DWORDs into the following channel context:

- CB, LLP, LIE, and RIE fields of the DMA Channel Control 1 register
- DMA Transfer Size
- DMA SAR Low and DMA SAR High
- DMA DAR Low and DMA DAR High

The definitions of the element DWORD bit fields are the same as the DMA Channel Context registers described in the “DMA Detail Register Description” section, with the exception of the LIE and RIE bits. The LIE and RIE bits in a LL element, only enable the done interrupt. In non-LL mode, the RIE and LIE bits (in the channel context registers) enable the done and abort interrupts.

10.6.4.4 Flow Control

10.6.4.4.1 Overview

This optional feature is available only for those channels which are configured to operate in

linked list (LL) mode. It enables your application to flow control the DMA controller by using PCIE_CLIENT_DMA_HSHAKE_TOGG register, that is, your application determines when data block transfer starts.

In normal mode, as soon as the DMA is doorbelled, the DMA reads the LL descriptor, and starts the data block transfer. When this feature is enabled, the DMA reads the descriptor, but starts the data transfer only when your application logic toggles the [w|r]dxfer_go_togg field in PCIE_CLIENT_DMA_HSHAKE_TOGG register.

The DMA handshake mechanism can be turned on or off per channel using

DMA_[WRITE/READ]_ENGINE_EN_OFF register. The handshake mechanism cannot be enabled/disabled when the channel status is active, that is, when DMA_CH_CONTROL1_OFF_[WR|RD]CH_i.CS =01.

10.6.4.4.2 DMA Handshake Operation

The handshake between the DMA and your application is done using [w|r]dxfer_go_togg and [w|r]dxfer_done_togg. In case of a DMA Write, your application should toggle the wdxfer_go_togg signal to indicate data block availability. In case of a DMA Read, your application should toggle the rdxfer_go_togg signal to indicate that your application hardware is ready to receive a data block.

To keep a track of [w|r]dxfer_go_togg signal toggles, the DMA implements a 5-bit handshake counter for each write/read channel. When your application toggles [w|r]dxfer_go_togg signal, the handshake counter is incremented. The handshake counter value is taken into account by the DMA before performing the data transfer for each descriptor. The data transfer happens only when the handshake counter value is non-zero. When the data transfer is complete, the DMA decrements the handshake counter, and toggles [w|r]dxfer_done_togg signal to indicate completion of data transfer to your application.

The handshake counter is 5-bit wide, so only 32 outstanding [w|r]dxfer_go_togg requests can be handled by the DMA for each channel. The DMA does not implement an overflow protection or overflow error indication mechanism for the handshake counter. It is the responsibility of your application to keep the number of [w|r]dxfer_go_togg toggles under check.

One example of the DMA operation when the handshake feature is enabled for a write channel is as follows:

1. The CPU doorbells channel 0.
2. The DMA resets the handshake counter for channel 0.
3. The DMA reads LL element 0 descriptor. After the DMA receives descriptor read completions, the DMA checks the value of the handshake counter.
If handshake counter =0, DMA waits until handshake counter >0 before transferring the data block.
If handshake counter >0, DMA transfers the data block immediately.
4. After the data transfer is complete, the DMA Decrement the handshake counter, and Toggles wdxfer_done_togg.
5. Steps 3-4 are repeated for all the remaining elements of the linked list.

10.6.4.5 Using the DMA

10.6.4.5.1 Source and Destination Address Registers

The DMA channel context SAR and DAR registers (DMA_SAR_LOW_OFF_WRCH_0, DMA_SAR_HIGH_OFF_WRCH_0 etc.) provide support for remote-to-local, and local-to-remote PCIe address mapping. You program the start of the local and remote data buffers using these registers, and the DMA increments the SAR and DAR as the DMA transfer progresses. For a write transfer, the SAR is the address of the local memory, and the DAR is the address of the remote memory, as shown in figure below.

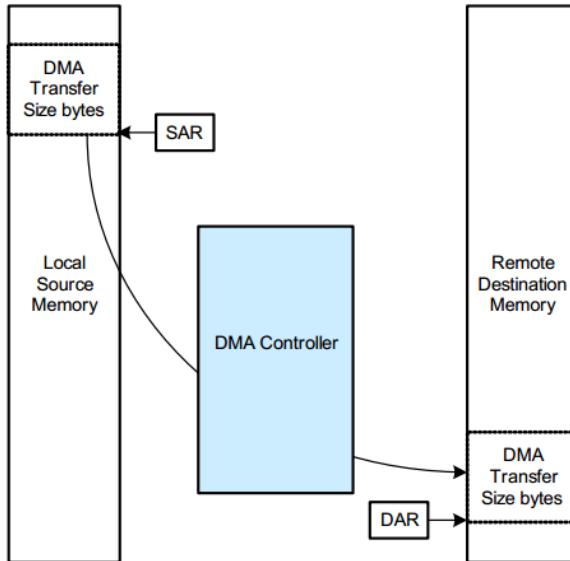


Fig. 10-15 Write Transfer: SAR and DAR for Write Channel

For a read transfer, the SAR is the address of the remote memory, and the DAR is the address of the local memory, as shown in Figure below.

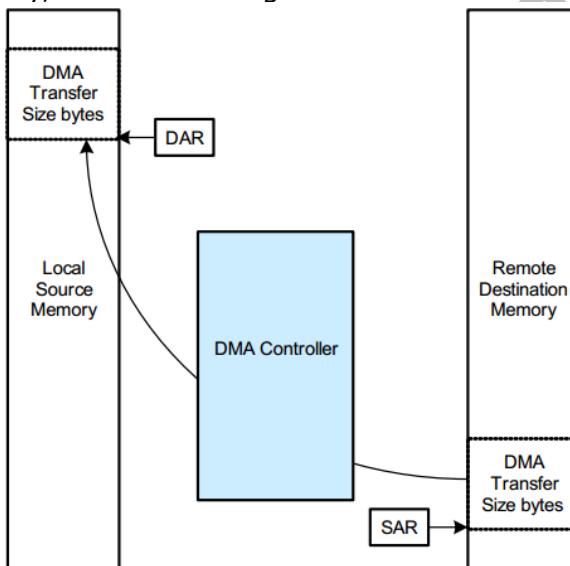


Fig. 10-16 Read Transfer: SAR and DAR for Write Channel

10.6.4.5.2 DMA Transfer Size Registers

You program the DMA transfer size using the DMA Transfer Size Register (DMA_TRANSFER_SIZE_OFF_WRCH_0 or DMA_TRANSFER_SIZE_OFF_RDCH_0). The maximum DMA transfer size is 4GB, and the minimum transfer size is one byte (0x1). The DMA decrements the value in this register as the DMA transfer progresses. When all bytes are successfully transferred, the value in this register is zero. In LL mode, the DMA overwrites this register with the corresponding dword of the LL element.

You can read this register to monitor the transfer progress. However, in some scenarios there is a delay before the controller updates this register. For example, when less than three channels are door belled, this register is only updated after a descriptor finishes (LL mode), or the transfer ends (non-LL mode).

10.6.4.5.3 Starting The DMA Transfer

After you program the DMA controller registers (including writing to the DMA Read Engine Enable or DMA Write Engine Enable register), you start a DMA transfer by writing zero to the Doorbell Number field of the DMA_WRITE_DOORBELL_0FF or DMA_READ_DOORBELL_0FF. You can program and start both a read and a write transfer at

the same time. The DMA supports full duplex operation, processing read and write transfers at the same time and in parallel with normal (non-DMA) traffic.

10.6.4.5.4 Detecting the End of The DMA Transfer

Detecting End of Transfer without Errors

The normal end of a DMA transfer is detected by any of the following methods:

- Local interrupt asserted.
- Remote interrupt (IMWr) received.
- Channel status field of the Channel Control 1 register is Stopped, and the DMA Transfer Size register is 0x0.
- Polling of the DMA Write Interrupt Status Register (DMA_WRITE_INT_STATUS_OFF) or DMA Read Interrupt Status Register (DMA_READ_INT_STATUS_OFF).

Detecting End of Transfer with Errors

The abnormal end of a DMA transfer is detected by any of the following methods:

- Local interrupt (pin) asserted.
- Remote interrupt (IMWr) received.
- Polling of the DMA Write Interrupt Status Register (DMA_WRITE_INT_STATUS_OFF) or DMA Read Interrupt Status Register (DMA_READ_INT_STATUS_OFF).
- Channel Status field of the Channel Control 1 register is Halted. When the DMA controller detects an error, it forces the DMA to stop issuing requests for the channel. It also sets the channel status field of the Channel Control 1 register to Halted, generates an abort interrupt (if enabled), and sends an abort IMWr (if enabled). The DMA Transfer Size register indicates the remaining number bytes to be transferred, except when there is an AXI write error during a DMA read transfer.
- Channel Status field of the Channel Control 1 register is Stopped, and the DMA Transfer Size register is not 0x0. You have prematurely stopped this channel as described in "Stopping the DMA Transfer (Software Stop)" on next section

10.6.4.5.5 Stopping the DMA Transfer (Software Stop)

You can manually abort (stop) the DMA transfer by writing the channel number to the Doorbell Number field and writing 1 to the Stop field in DMA_[WRITE|READ]_DOORBELL_OFF. This causes the DMA to:

- Place the channel in a Stopped state. The channel Status field in DMA_CH_CONTROL1_OFF_WRCH_0 is Stopped and the value in DMA_TRANSFER_SIZE_OFF_WRCH_0 will not be 0x0.
- Wait for all outstanding pending transactions.
- Assert the abort interrupt (if it is enabled) in DMA_WRITE_INT_STATUS_OFF.

You might also do this as part of a function level reset (FLR). FLR does not directly affect the DMA transfer so you must manually stop the DMA transfer before initiating an FLR.

10.6.4.6 Programming Examples

This section provides two programming and operation example flows.

10.6.4.6.1 Non-LL Write Transfer

In this example, the IMWr generation is disabled, as the local CPU initiates the DMA transfer. The local CPU is interrupted using the pcie_sys_int interrupt. The SAR is the address of the local memory, and the DAR is the address of the remote memory. Write DMA Transfer is from 0xBEEF_BEE0 to 0xCAFE_CAF0. Table below provides the programming details for this example transfer. The transfer size is 1MB.

Table 10-7 Write DMA Transfer Example

| Address 0x380000+ | NAME | Value |
|----------------------|---|-------|
| 0x00C | DMA Write Engine Enable You must not write 0 to this register. Even temporarily writing 0 to this register resets the DMA logic. | 0x1 |

| Address 0x380000+ | NAME | Value |
|----------------------|---|-------------|
| 0x054 | DMA Write Interrupt Mask | 0x0 |
| 0x200 | DMA Channel Control 1 register ■ Local Interrupt Enable (LIE) =1 ■ Remote Interrupt Enable (RIE) =0 ■ AT, RO, NS, TC, Function Number =0 | 0x04000008 |
| 0x208 | DMA Transfer Size | 0x00100000 |
| 0x20C | DMA SAR Low | 0xBEEF_BEE0 |
| 0x210 | DMA SAR High | 0x0000_0000 |
| 0x214 | DMA DAR Low | 0xCAFE_CAF0 |
| 0x218 | DMA DAR High | 0x0000_0000 |
| 0x010 | DMA Write Doorbell | 0x0 |

10.6.4.6.2 Non-LL Read Transfer

In this example, the local interrupt generation is disabled, as the remote CPU initiates the DMA transfer. The remote CPU is interrupted using an IMWr. The SAR is the address of the remote memory, and the DAR is the address of the local memory. Read DMA Transfer from 0xBEEF_BEE0 to 0xCAFE_CAF0. Table below provides the programming details for this example transfer. The transfer size is 1MB.

Table 10-8 Read DMA Transfer Example

| Address 0x380000+ | NAME | Value |
|----------------------|---|-----------------|
| 0x02C | DMA Read Engine Enable You must not write 0 to this register. Even temporarily writing 0 to this register resets the DMA logic. | 0x1 |
| 0x0CC/0x0D0 | DMA Read Done IMWr Address Low and high | IMWr Address #1 |
| 0x0D4/0x0D8 | DMA Read Abort IMWr Address Low and High | IMWr Address #2 |
| 0x0DC | DMA Read Channel 0 IMWr Data | IMWr Data |
| 0x200 | DMA Channel Control 1 register ■ Local Interrupt Enable (LIE) =0 ■ Remote Interrupt Enable (RIE) =1 ■ AT, RO, NS, TC, Function Number =0 | 0x04000010 |
| 0x208 | DMA Transfer Size | 0x00100000 |
| 0x20C | DMA SAR Low | 0xBEEF_BEE0 |
| 0x210 | DMA SAR High | 0x0000_0000 |
| 0x214 | DMA DAR Low | 0xCAFE_CAF0 |
| 0x218 | DMA DAR High | 0x0000_0000 |
| 0x030 | DMA Read Doorbell | 0x0 |

10.6.5 PCIe Message Handling

This section describes the processing of messages through the controller. For a proper understanding of messages you should be familiar with Section 2.2.8, "Message Request Rules" of the PCI Express Base Specification, Revision 3.0.

10.6.5.1 Message Generation

These are three methods to generate PCIe Messages:

- Specific MSG interface
You can use PCIE_CLIENT_MSG_GEN_CON register to generate messages such as:
 - Vendor-Defined Message
 - Legacy PCI Interrupt Message
 - PME_Turn_Off message
 - Unlock message
 - LTR Message
 - OBFF Message
 Refer to register description for more details.
- AXI bridge
In this method, you should program the PCIE_CLIENT_AXI_SLV_AWMISC_HDR, PCIE_CLIENT_AXI_SLV_AWMISC_HDR3 and PCIE_CLIENT_AXI_SLV_AWMISC_HDR4 registers to set these parameters:
`slv_awmisc_info [4:0] =MSG`
`slv_awmisc_info [20:13] = message code`
`slv_awmisc_tag [5:0] = tag`
`slv_awmisc_info_hdr_[3|4]dw = 3rd and 4th TLP header DWORD`
 Then you can send zero length memory write (`slv_awstrb=0`) to generate the message.
 You must not program registers mentioned above when there are normal transfers pending on AXI slave interface. And should set these registers in normal value when you want to send MWr TLP. This method is not recommend because of these limitations.
- AXI bridge and iATU
Program related client register fields as follows:
`slv_awmisc_info[4:0] =MEM`
`slv_awmisc_info_hdr_[3|4]dw = 3rd and 4th TLP header DWORD`
 Then iATU needs to be configured to translate MWr to Msg TLPs. Refer to "Address Translation" for more details. This method is not recommend because that it needs additional iATU resources.

10.6.5.2 Message Reception

PCIe controller use `pcie_msg_int` interrupt to indicate the reception of PCIe Message. Application can poll to `PCIE_CLIENT_INTR_STATUS_MSG_RX` register to check the status of Message reception. Refer to register information for details.

10.6.6 PCIe Power Management

10.6.6.1 Overview

The controller supports two categories of PM operations to control the device state (D-state) and link state. For a proper understanding of PCIe Power management you should be familiar with Section 5, "Power Management" of the PCI Express Base Specification, Revision 3.0.

Software PCI Compatible PM (PCI-PM)

- D-state PM of Function. The host software can direct the function to enter any of the D1, D2, or D3 low-power states. It does this by writing to the Power Management Control and Status Register (PMCSR) in the PCI-PM capability structure.
- D-state PM of Link. Link states are not visible to PCI-PM legacy compatible software, and are derived from the power management D-states of the components connected to that link. The action of changing the D-state in the PMCSR indirectly causes a change in the link power state. The L1 state is entered whenever all functions of a USP on a link are programmed to a non-D0 state. The entry into L2 and L3 states is initiated by the DSP.

PCIe PM Mechanisms

- Active State PM (ASPM). When the USP is in L0 and detects idleness on the link for a specific amount of time, it automatically transitions the link to the L0s or L1 (optional) power state.
- L1 Substates. This is an optional PCIe feature that enables components on a link to further reduce idle power consumption while the link is in L1, including almost complete

- removal of power for the high speed PHY circuits.

10.6.6.2 L0s Operation

L0s is a low-power state enabled by ASPM. ASPM controls entry into L0s for the transmitter. The remote device controls entry into L0s for the receiver.

10.6.6.2.1 L0s Entry

All of these condition must be met:

- ASPM L0s is enabled through the ASPM Control field in the Link Control register.
- L0s entry conditions as defined in Section 5.4.1.1.1, “Entry into the L0s State” of the PCI Express Base Specification, Revision 3.0, exist for a duration of time (determined by the LOS_ENTRANCE_LATENCY field in ACK_F_ASPM_CTRL_OFF).
- No higher stage of power-down requested.

10.6.6.2.2 L0s Exit

Any of these condition can be met:

- A DLLP or TLP is pending to be sent.
- L1 entry conditions as defined in Section 5.4.1.2.1, “Entry into the L1 State” of the PCI Express Base Specification, Revision 3.0, are satisfied.
- PCIe link partner requests to enter into link recovery.

10.6.6.3 L1 Operation (Non-substates)

The following topic will be discussed in this section: L1 (ASPM/PM) Entry and Exit Conditions and L1 Clock PM (L1 with REFCLK removal/PLL Off) Entry and Exit Conditions.

10.6.6.3.1 L1 (ASPM/PM) Entry

L1 is a low-power state enabled either by ASPM (L1-ASPM) or by the software changing the D-state (L1-PM). The L1 state is a bi-directional link low-power state and both link partners must negotiate to go to this state.

The L1-ASPM entry negotiation handshake uses PM_Active_State_Request_L1 DLLPs, PM_Request_Ack DLLPs, and PM_Active_State_Nak MSG TLPs. Refer to PCIe Basic Specification for more information.

There are three scenarios that cause the controller to enter L1 under ASPM conditions.

Scenario 1: L1 Idle Timeout In L0s

All of these condition must be met in the USP:

- ASPM L1 and L0s are enabled through the ASPM Control field in the Link Control register.
- The ENTER_ASPMfield of ACK_F_ASPM_CTRL_OFFis set to ‘0’ and the link state is L0s for both link partners, or the ENTER_ASPMfield of ACK_F_ASPM_CTRL_OFFis set to ‘1’.
- L1 entry conditions as defined in Section 5.4.1.2.1, “Entry into the L1 State” of the PCI Express Base Specification, Revision 3.0, exist for a duration of time (determined by the L1_ENTRANCE_LATENCYfield in ACK_F_ASPM_CTRL_OFF).
- No higher stage of power-down requested.
- Your USP application is not asserting the app_xfer_pending field in PCIE_CLIENT_POWER_CON register.
- There are no in-progress transactions in the controller
- There are no pending requests at the AXI slave interface, that is, slv_a*valid must be 0.
- There are no pending DMA transfers.

Scenario 2: L1 Idle Timeout In L0

All of these condition must be met in the USP:

- ASPM L1 is enabled and L0s is not enabled through the ASPM Control field in the Link Control register.
- Link state is L0.
- L1 entry conditions as defined in Section 5.4.1.2.1, “Entry into the L1 State” of the PCI Express Base Specification, Revision 3.0, exist for a duration of time (determined by the L1_ENTRANCE_LATENCYfield in ACK_F_ASPM_CTRL_OFF).
- No higher stage of power down-requested.

- Your USP application is not asserting the app_xfer_pending field in PCIE_CLIENT_POWER_CON register.
- There are no in-progress transactions in the controller.
- There are no pending requests at the AXI slave interface, that is, slv_a*valid must be 0.
- There are no pending DMA transfers.

Scenario 3: Application Controlled (USP only)

All of these condition must be met in the USP:

- ASPM L1 is enabled through the ASPM Control field in the Link Control register.
- Your application write 1 to the app_req_entr_l1 field in PCIE_CLIENT_POWER_CON register.
- Your USP application is not asserting the app_xfer_pending field in PCIE_CLIENT_POWER_CON register.
- There are no in-progress transactions in the controller.
- There are no pending requests at the AXI slave interface, that is, slv_a*valid must be 0.
- There are no pending DMA transfers.

L1-PM Entry

The power management state of a link is determined by the D-state of the USP. When you change the device state of the USP to D1, D2, or D3hot by writing to the PMCSR, the controller must initiate a link state transition to L1.

An USP application asserting the app_xfer_pending field does not prevent L1-PM entry, but will cause immediate exit from L1-PM.

10.6.6.3.2 L1-PM/L1-ASPM Exit

Any of these condition are met:

- A DLLP or TLP is pending to be sent.
- Your application asserts the app_req_exit_l1 field.
- Link partner is requesting exit from L1.
- Your application asserts the app_xfer_pending field.
- Your application asserts the app_pm_xmt_pme field.

When the USP is programmed with capability to support PME; it sends a PME message to the RC which calls the PM software to transition the USP to the D0 state. Therefore, you should only use app_pm_xmt_pme for L1-PM exit.

- PM software (RC) requests a higher stage of power-down by writing to the PMCSR in the USP.
- Your application (USP) requests transmission of VDM, MSI/MSIX, or LTR message.

Legacy interrupt is not included.

- Your application (DSP) requests transmission of Unlock message.
- Your application is requesting to send traffic by asserting or slv_a*valid.
- Your application doorbells a DMA read or write channel, or DMA controller is requesting data.

■ Your application (DSP) initiates link disable, or link retrain (by setting PCIE_CAP_LINK_DISABLE or PCIE_CAP_RETRAIN_LINK field in LINK_CONTROL_LINK_STATUS_REG to 1).

- Your application (DSP) initiates hot reset by either:

- setting RESET_ASSERT field in PORT_LINK_CTRL_OFF to 1, or
 - setting SBR field in BRIDGE_CTRL_INT_PIN_INT_LINE_REG to 1, or
 - toggling app_init_rst field in PCIE_CLIENT_GENERAL_CON.

- Your application requests a speed change (by setting DIRECT_SPEED_CHANGE field in GEN2_CTRL_OFF to 1).

- Your application requests link width change (by setting DIRECT_LINK_WIDTH_CHANGE field in MULTI_LANE_CONTROL_OFF is set to 1).

10.6.6.3.3 L1 Clock PM

For an USP, host software uses the Enable Clock Power Management bit in the Link Control register to enable this feature. For a DSP, this register bit is hard coded to '0' and cannot be

used to control this behavior. Your application can use the app_clk_pm_en input to dynamically control whether to execute L1 with or without Clock PM.

■ The Support Clock Power Management bit in the Link Capabilities register must be set. For downstream ports it is hardcoded to 0, for upstream ports it can be accessed through the DBI.

■ The Enable Clock Power Management bit in the Link Control register must be set.

■ You must set the app_clk_pm_en field to 1. The controller only samples app_clk_pm_en when L1 is entered.

L1 with Clock PM and L1 substates work orthogonal to each other. L1 with Clock PM uses the mac_phy_pclkreq_n[0]signaling, and L1 substates uses the mac_phy_pclkreq_n[1]signaling. However, L1 substates takes precedence over Clock PM within the cores PM state machine.

10.6.6.4 L1 Substate

The L1 substates are applicable in both the ASPM and PCI-PM L1 link states. L1 substates management utilizes a per-link sideband signal called CLKREQ#.

■ During L1 substates it is assumed that core_clk is turned off and that aux_clk is active. It is required that your application switches aux_clk to a low frequency free running clock on entry into L1.

■ The controller uses aux_clk for counting time during L1 substates. You must program the frequency of this clock into the L1_SUBSTATES_OFF register with a value in the range 1...1000MHz to count real time. Frequencies lower than 1 MHz are possible, but with a loss of accuracy in the time counted.

■ When the electrical idle detection circuitry is disabled it is assumed that the PHY holds the signal phy_mac_rxidle to 1.

After the link has entered L1 through the normal L1 negotiation, the USP can initiate the sequence for entering the target L1 substate (L1.1 or L1.2) by tri-stating its CLKREQ# output buffer. The entry sequence can only proceed if the DSP is also tri-stating its CLKREQ# output buffer, resulting in the bidirectional CLKREQ# signal being pulled up to 1. Otherwise CLKREQ# will remain asserted at 0 and the link state will stay in L1. The exit sequence can be initiated by either ports by asserting CLKREQ# to 0. For each port there are two cases to consider, the first where the exit is initiated locally, the second where the exit is initiated remotely. L1 substates management utilizes a per-link sideband signal called CLKREQ#.

10.6.6.4.1 L1 Substates Software Control

When the controller enters L1, the target L1 substate depends on several programming bits:

- PM Control/Status Register
 - USP: The current D-state
 - DSP: The DLLP type that was used by USP to request L1
- L1 Substates Control 1 Register
 - ASPM PM L1.1 Enabled
 - ASPM PM L1.2 Enabled
 - PCI PM L1.1 Enabled
 - PCI PM L1.2 Enabled

After the USP controller enters L1, it uses the D-state of the device to determine if L1 was entered in ASPM mode or PCI-PM mode.

After the DSP controller enters L1, it uses the USPs DLLPL1 request type to determine if L1 was entered in ASPM mode or PCI-PM mode. The next tables shows the target L1 substate as a function of the relevant programming bits.

Table 10-9 Target L1 Substate as a Function of Software Controls (USP)

| D-State | ASPM L1.1 Enabled | ASPM L1.2 Enabled | LTR >= Threshold | PCI PM L1.1 Enabled | PCI PM L1.2 Enabled | Target L1 Substate |
|---------|-------------------|-------------------|------------------|---------------------|---------------------|--------------------|
| !D0 | - | - | - | 0 | 0 | L1 |
| !D0 | - | - | - | 1 | 0 | L1.1 |
| !D0 | - | - | - | - | 1 | L1.2 |
| D0 | 0 | 0 | - | - | - | L1 |
| D0 | 1 | 0 | - | - | - | L1.1 |
| D0 | 0 | 1 | 0 | - | - | L1 |
| D0 | 1 | 1 | 0 | - | - | L1.1 |
| D0 | - | 1 | 1 | - | - | L1.2 |

Table 10-10 Target L1 Substate as a Function of Software Controls (DSP)

| DLLP Receive | ASPM L1.1 Enabled | ASPM L1.2 Enabled | LTR >= Threshold | PCI PM L1.1 Enabled | PCI PM L1.2 Enabled | Target L1 Substate |
|----------------------------|-------------------|-------------------|------------------|---------------------|---------------------|--------------------|
| PM_Enter_L1 | - | - | - | 0 | 0 | L1 |
| | - | - | - | 1 | 0 | L1.1 |
| | - | - | - | - | 1 | L1.2 |
| PM_Active_State_Request_L1 | 0 | 0 | - | - | - | L1 |
| | 1 | 0 | - | - | - | L1.1 |
| | 0 | 1 | 0 | - | - | L1 |
| | 1 | 1 | 0 | - | - | L1.1 |
| | - | 1 | 1 | - | - | L1.2 |

The Reported LTR is the maximum of the snoop/no snoop latency values embedded in LTR messages transmitted by the upstream ports or received by the downstream port. The controller stores these values in the port logic LTR Latency Register (LTR_LATENCY_OFF). When the requirement bit in the message is 0, the latency value is considered infinite (that is, the check with the threshold always pass).

10.6.6.4.2 L1 Substates Entry and Exit

PCIe link entered L1 substate automatically if related enable bit is set and LTSSM already in L1 state, refer to section L1 Substates Software Control for details.

L1 substates Exit can be triggered locally or Remotely. For locally initiated exit, refer to section L1-PM/L1-ASPM Exit. Remotely initiated exit begin when CLKREQ# is asserted by remote PCIe partner.

10.6.6.5 L2 and L3 Power Down Entry and Exit

L2/L3 entry is initiated after the RC calls power management software to initiate the removal of power and clocks. USPs of devices in D0, D1, D2, and D3hot must respond to the receipt of a PME_Turn_Off MSG TLP by transmitting a PME_TO_Ack MSG TLP. The device must then request a link transition to L2/L3_Ready. L2/L3_Ready is a bi-directional link power down state. If your application is not ready to be shut-down, it must keep the app_ready_entr_l23 field de-asserted.

L2/L3 Entry

All of these condition must be met:

- PME_Turn_Off/PME_TO_Ack handshake has been completed.
- Your USP application is ready to be turned off; app_ready_entr_l23=1.
- After sending the PME_TO_Ack, the USP initiates the L2/L3 Ready transition protocol by sending the PM_Enter_L23 DLLP. The RC responds with the PM_Request_Ack.

L2/L3 Exit

Any of these condition can be met:

- When the USP is programmed with capability to support PME, your application can assert the apps_pm_xmt_pme field to request the controller to wake up. The USP then sends a PM_PME MSG TLP to the RC which calls the PM software to transition the USP out of the D3 state.
- Device is programmed with capability to support PME and your application requests the controller to wake up by triggering a native hot-plug event.
- Link partner is requesting exit from L2/L3.

10.6.6.6 Dynamic Power Allocation (DPA)

The DPA capability enables software to actively manage and optimize function power usage when in the D0 state. DPA is not applicable to power states D1-D3.

For details on how your application interacts with the controller, see the description of the dpa_sub_upd_int field in PCIE_CLIENT_INTR_STATUS_MISC register.

10.6.7 PCIe Interrupt

The PCIe provides six types of interrupt to system interrupt controller. They can be divided into MSI/MSI-X, PCIe Error interrupt, PCIe Message Receive interrupt, PCIe Legacy interrupt, PCIe System interrupt and PCIe Power Management Interrupt. When operating as RC, the PCIe is capable of handling both MSI/MSI-X and legacy interrupts. This is because when operating as RC it should be able to service both PCIe end points as well as legacy endpoints. It is capable of generating MSI or Legacy interrupt if the PCIe is configured as EP. Notes that PCIe EP component can't generate both Legacy and MSI/MSI-X interrupt. It is either one or the other. The interrupt type an EP generates is configured during configuration time.

Interrupt status and mask bits are located in client register group, some interrupts are handle by client register directly, but some interrupts are generated by events deep into controller and software should clear the root cause to serve the interrupt events. For more information, refer to the register description.

Table 10-11 PCIe Interrupt Table

| System Interrupt Event ID | Interrupt description (level 1) | Interrupt subset (level 2) | Support mode |
|----------------------------------|--|-------------------------------------|---------------------|
| 118 | PCIe System Interrupt | PHY link up interrupt | RC & EP |
| | | DLL link up interrupt | RC & EP |
| | | Link down reset request interrupt | RC & EP |
| | | Slot status change interrupt | RC |
| | | Hot plug interrupt | RC |
| | | Link autonomous bandwidth interrupt | RC |
| | | Bandwidth Management Interrupt | RC |
| | | EDMA write channel interrupt | RC & EP |
| | | EDMA read channel interrupt | RC & EP |
| | | DPA update interrupt | EP |
| 119 | PCIe Legacy interrupt | Resizable BAR update interrupt | EP |
| | | INTA received interrupt | RC |
| | | INTB received interrupt | RC |
| | | INTC received interrupt | RC |
| | | INTD received interrupt | RC |
| | | INTA sent interrupt | EP |
| | | INTB sent interrupt | EP |
| | | INTC sent interrupt | EP |
| 120 | PCIe message received interrupt | INTD sent interrupt | EP |
| | | Vendor message received interrupt | RC & EP |
| | | Unlock message received interrupt | EP |
| | | LTR message received interrupt | RC |
| | | PME status Interrupt | RC |

| System Interrupt Event ID | Interrupt description (level 1) | Interrupt subset (level 2) | Support mode |
|---------------------------|---------------------------------|--|----------------------------|
| | | PM_PME message received interrupt | RC |
| | | PME_TO_Ack message received interrupt | RC |
| | | PME Turnoff message received interrupt | EP |
| | | 'IDLE' OBFF message received interrupt | RC |
| | | 'OBFF' OBFF message received interrupt | RC |
| | | 'CPU Active' OBFF message received interrupt | RC |
| 121 | PCIe Error interrupt | Root Error Status interrupt | RC |
| | | Completion TLP Rx timeout interrupt | RC & EP |
| | | Completion TLP Tx timeout interrupt | RC & EP |
| | | ERR_COR message sent interrupt | EP |
| | | ERR_NONFATAL message sent interrupt | EP |
| | | ERR_FATAL message sent interrupt | EP |
| | | ERR_COR message received | RC |
| | | ERR_NONFATAL message received | RC |
| | | ERR_FATAL message received | RC |
| | | Receive FIFO overflowed interrupt | RC & EP |
| 136 | PCIe Power Management interrupt | L1 substate entry interrupt | RC & EP |
| | | L1 entry interrupt | RC & EP |
| | | L2 entry interrupt | RC & EP |
| | | L0s entry interrupt | RC & EP |
| | | L1 substate exit interrupt | RC & EP |
| | | L1 exit interrupt | RC & EP |
| | | L2 exit interrupt | RC & EP |
| | | L0s exit interrupt | RC & EP |
| | | D-state changed interrupt | EP |
| XXX | MSI/MSI-X | message signaled interrupt | RC service and EP initiate |

10.7 PCIe Appendix

10.7.1 Programming setting for PCIe PHY TX PLL

10.7.1.1 Reference Clock Frequency 24MHz

24MHz is the default setting after reset, and parameter values related is as follows:

32'h2118 address write 32'h02

32'h211c address write 32'h71

32'h2020 address write 32'h86

32'h2028 address write 32'h71

32'h2030 address write 32'h26

32'h3020 address write 32'h86

32'h3028 address write 32'h71

32'h3030 address write 32'h26

10.7.1.2 Reference Clock Frequency 25MHz

32'h2118 address write 32'h00
32'h211c address write 32'h64
32'h2020 address write 32'h01
32'h2028 address write 32'h64
32'h2030 address write 32'h21
32'h3020 address write 32'h01
32'h3028 address write 32'h64
32'h3030 address write 32'h21

10.7.1.3 Reference Clock Frequency 50MHz

32'h2118 address write 32'h00
32'h211c address write 32'h32
32'h2020 address write 32'h01
32'h2028 address write 32'h32
32'h2030 address write 32'h21
32'h3020 address write 32'h01
32'h3028 address write 32'h32
32'h3030 address write 32'h21

10.7.1.4 Reference Clock Frequency 100MHz

32'h2118 address write 32'h0;
32'h211c address write 32'h19;
32'h2020 address write 32'h01;
32'h2028 address write 32'h19;
32'h2030 address write 32'h21;
32'h3020 address write 32'h01;
32'h3028 address write 32'h19;
32'h3030 address write 32'h21;

Chapter 11 Voice Activity Detect (VAD)

11.1 Overview

Voice Activity Detect(VAD) is used to detect the amplitude of voice which is received by Analog Mic, I2S Digital Mic or PDM digital Mic when SoC is in low power mode. If the amplitude of voice is over threshold, the VAD will assert interrupt to wake up SoC, then SoC will exit low power mode.

VAD supports the following features:

- Support AHB bus interface
- Support read voice data from I2S_8CH_0,I2S_2CH_1,PDM
 - Support to configure the voice source address
 - Support to configure increment or fixed for the direction of voice data address
 - Support 3 group of DMA request and acknowledge
 - Support transfer 1~8 burst per DMA request
 - Support read 1~8 Mic voice data, and only support single Mic voice detection, user can select any Mic voice data to detect the amplitude of voice
 - Support 16/24 bits voice data
- Support voice amplitude detection
 - Support an Amplifier for the voice data
 - Support a IIR high pass filter for the voice frequency band, and the filter coefficient can be configured
 - Support a voice detect threshold that take the ambient noise to account
- Support Multi-Mic array data storing
 - Buffer memory is shared with Internal SRAM
 - The start and end address of storing can be configured
 - When current storing address is up to end address, it will loop to start address and overlap previous data, it will also assert a flag
 - Support 3 data storing mode: mode 0 start storing data after the voice detect event, mode 1 start storing after VAD is enabled and mode 2 do not storing data
- Support a level combined interrupt
 - Support voice detect interrupt
 - Support time out interrupt
 - Support transfer error interrupt
 - Support data transfer interrupt

11.2 Block Diagram

VAD comprises with:

- ahb_master: AHB Master Interface
- ahb_sram_if: AHB Slave Interface
- vad_reg_bank : Register bank
- dmac_engine: DMA control engine
- vad_det : Voice detection

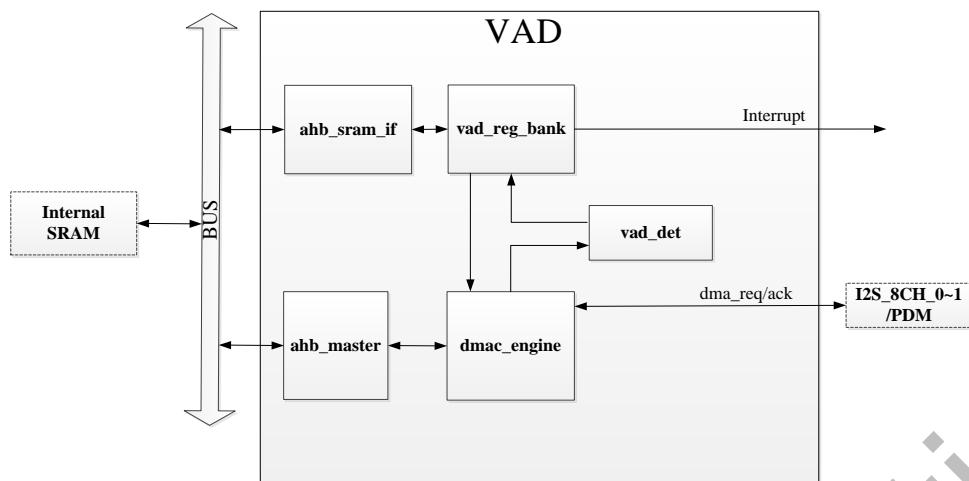


Fig. 11-1 VAD Block Diagram

11.3 Function Description

11.3.1 DMAC_ENGINE

dmac_engine is used to read voice data from one of I2S_8CH_0,I2S_2CH_1 or PDM, and it can store all channels data to Internal SRAM. The voice data can be 16 or 24 bits:

- When it's 16 bits, it must be half word transfer mode that low 16 bits in a word for left channel and high 16 bits in a word for right channel.
- When it's 24bits, it must be word transfer mode that only 24 bits data is valid in a word, and it support left or right justified

dmac_engine also select and send one channel data to vad_det for voice detection. vad_det only support 16 bits data to detect the amplitude of voice, so when the voice data is 24 bits, user can use the high or low 16 bits in 24 bits.

- When use high bits, the data value will be divided by 256.
- When use low bits, the data value will be saturation to 16 bits.

11.3.2 VAD_DET

vad_det is used to detect the amplitude of voice.

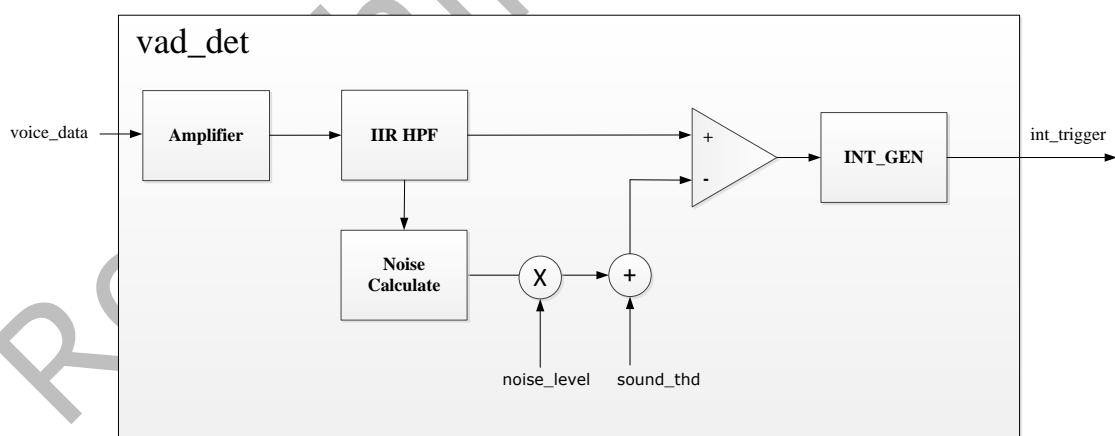


Fig. 11-2 VAD_det Block Diagram

Amplifier

$\text{voice_amplitude_amplified} = \text{gain} * \text{voice_amplitude_original} / 8$.

IIR HPF

There is a high pass filter for the human voice frequency band, the filter is a two order direct I type IIR. The following formula describes:

$$y(n) = -a1 * y(n-1) - a2 * y(n-2) + b0 * x(n) + b1 * x(n-1) + b2 * x(n-2)$$

The coefficient a1, a2, b0, b1 and b2 are all quantified by multiplying 16384 and

represented as 16 bits, the result is follow registers that can be configured: iir_anum_0, iir_anum_1, iir_anum_2, iir_aden_1 and iir_aden_2.

The output of HPF need some time to achieve convergence after VAD is enabled.

Noise Calculate

VAD support a voice detection threshold that take the ambient noise to account:

- VAD calculate the average amplitude of voice data within noise_sample_num samples, the result is regard as the noise value of one frame. The noise value of last 128 frames also can be configured directly.
- VAD find the minimum noise value within noise_frm_num frames, the result is regard as the noise_min(minimum noise value).User can configure min_noise_find_mode to change the mode to find the minimum noise.
- The noise_min will be smooth updated to noise_abs(current noise value), the formula is as follow: $\text{noise_abs} = (\text{noise_abs} * \text{noise_alpha} + \text{noise_min} * (256 - \text{noise_alpha})) / 256$. noise_abs will be updated once every frame. noise_abs also can be configured directly, and it is not clear until VAD is reset.

Voice Detect Threshold

The final threshold is sound_thd + noise_abs * noise_level.

INT_GEN

VAD support 3 modes to assert the voice detection interrupt.

- Normal mode: When equal or more than a number (vad_con_thd) of continuous samples over the threshold, the voice detection interrupt will be asserted. The vad_con_thd can be configured by register.
- Allow an exception mode: base on normal mode, it can be configured to allow exceptions during continuous sample judgment. The exception number can be configured by register. When the exceptions is more, the voice detect condition is less strict.
- Accumulating mode: A counter is used for accumulating, the counter will plus 1 when current sample is over threshold (it will not plus when it reach maximum value 256), the counter will minus 1 when current sample is not over threshold (it will not minus when it reach 0); When the counter is equal or more than a number (vad_con_thd), the voice detection interrupt will be asserted. Compare with normal mode, the voice detect condition is less strict when use the same value of vad_con_thd.

11.4 Register Description

11.4.1 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|--------------------|--------|------|-------------|-------------------------------|
| VAD CONTROL | 0x0000 | W | 0x03000000 | Control register |
| VAD VS ADDR | 0x0004 | W | 0x00000000 | Voice source address register |
| VAD TIMEOUT | 0x004c | W | 0x00000000 | Timeout register |
| VAD RAM START ADDR | 0x0050 | W | 0x00000000 | RAM start address register |
| VAD RAM END ADDR | 0x0054 | W | 0x00000000 | RAM end address register |
| VAD RAM CUR ADDR | 0x0058 | W | 0x00000000 | RAM current address register |
| VAD DET CON0 | 0x005c | W | 0x01024008 | Detect control register0 |
| VAD DET CON1 | 0x0060 | W | 0x04ff0064 | Detect control register1 |
| VAD DET CON2 | 0x0064 | W | 0x3bf5e663 | Detect control register2 |
| VAD DET CON3 | 0x0068 | W | 0x3bf58817 | Detect control register3 |
| VAD DET CON4 | 0x006c | W | 0x382b8858 | Detect control register4 |
| VAD DET CON5 | 0x0070 | W | 0x00000000 | Detect control register5 |
| VAD INT | 0x0074 | W | 0x00000000 | VAD Interrupt register |
| VAD AUX CONTROL | 0x0078 | W | 0x00000000 | Auxiliary control register |
| VAD SAMPLE CNT | 0x007c | W | 0x00000000 | Sample counter register |
| VAD NOISE DATA | 0x0100 | W | 0x00000000 | Noise data register |

Notes: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

11.4.2 Detail Register Description

VAD CONTROL

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:29 | RW | 0x0 | vad_det_channel Index of the channel for voice detect, from channel 0 to channel 7 |
| 28 | RW | 0x0 | voice_24bit_sat The mode of voice 24bit data change to 16bit 1'b0: Get the high 16bit data(divided by 256) 1'b1: Saturation from 24bit to 16bit |
| 27 | RW | 0x0 | voice_24bit_align_mode Align mode of channel 24bit width 1'b0: 8~31bits is valid 1'b1: 0~23bits is valid |
| 26 | RW | 0x0 | voice_channel_bitwidth 1'b0: 16bits 1'b1: 24bits |
| 25:23 | RW | 0x6 | voice_channel_num Voice channel number, the value N means N+1 channel |
| 22 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 21:20 | RW | 0x0 | vad_mode 2'h0: Begin to store the data after voice detect 2'h1: Begin to store the data after VAD is enable 2'h2: Don't store the data 2'h3: Reserved |
| 19:15 | RO | 0x00 | Reserved |
| 14 | RW | 0x0 | source_fixaddr_en Direction of source address 1'b0: Increment 1'b1: Fixed |
| 13:10 | RW | 0x0 | incr_length INCR burst length, 0~15 is valid It is valid when source_burst is set to 3'h1 |
| 9:7 | RW | 0x0 | source_burst_num Source burst number per dma_req, the value N means N+1 burst |
| 6:4 | RW | 0x0 | source_burst 3'h0: SINGLE 3'h1: INCR 3'h3: INCR4 3'h5: INCR8 3'h7: INCR16 Others: Reserved |
| 3:1 | RW | 0x0 | source_select Voice source select 3'h1: I2S0_8CH 3'h3: I2S1_2CH 3'h4: PDM Others: Reserved |
| 0 | RW | 0x0 | vad_en VAD enable 1'b0: Disable 1'b1: Enable |

VAD VS ADDR

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------|
| 31:0 | RW | 0x00000000 | vs_addr Voice source address |

VAD TIMEOUT

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | work_timeout_en Work timeout enable 1'b0: Disable 1'b1: Enable |
| 30 | RW | 0x0 | idle_timeout_en Idle timeout enable 1'b0: Disable 1'b1: Enable |
| 29:20 | RW | 0x000 | work_timeout_thd Work timeout threshold, the unit is one cycle of hclk |
| 19:0 | RW | 0x00000 | idle_timeout_thd Idle timeout threshold, the unit is one cycle of hclk |

VAD RAM START ADDR

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | ram_begin_addr RAM start address to store voice data, the address must be double word alignment |

VAD RAM END ADDR

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | ram_end_addr RAM end address to store voice data, the address must be double word alignment |

VAD RAM CUR ADDR

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:0 | RW | 0x00000000 | ram_cur_addr RAM current address to store voice data, The last valid double word data is at address ram_cur_addr-0x8. When the ram_loop_flag is valid, the valid voice data will be ram_cur_addr ~ ram_end_addr ~ loop to ram_begin_addr ~ ram_cur_addr-0x8. When the ramp_loop_flag is not valid, the valid voice data will be ram_begin_addr ~ ram_cur_addr-0x8 |

VAD DET CON0

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:30 | RO | 0x0 | Reserved |
| 29:28 | RW | 0x0 | vad_thd_mode Threshold mode for vad_con_thd 2'b00: Normal mode 2'b01: Allow an exception mode 2'b10: Accumulating mode 2'b11: Reserved |
| 27:24 | RW | 0x1 | dis_vad_con_thd In the determining of continuous sample number exceed threshold, allow some number of sample as an exception. It's valid only when vad_thd_mode=1. When this value is lower, the voice detect condition is more strict |
| 23:16 | RW | 0x02 | vad_con_thd When continuous sample number(>=vad_con_thd) exceed threshold, then assert the vad_det interrupt, the value N means N+1. When this value is higher, the voice detect condition is more strict |
| 15 | RO | 0x0 | Reserved |
| 14:12 | RW | 0x4 | noise_level Noise level, valid value is 0x1~0x6 When this value is higher, the voice detect condition is more strict |
| 11:0 | RW | 0x008 | gain The gain control of voice data amplifier, the value of gain is unsigned and is valid from 0 to 4095. voice_amplitude_amplified=gain*voice_amplitude_original/8. |

VAD DET CON1

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31 | RO | 0x0 | Reserved |
| 30 | RW | 0x0 | min_noise_find_mode Minimal noise value find mode 1'b0: Always find the value at the range of noise_frm_num 1'b1: When receive N frame: if N is less than noise_frm_num, find the value at the range of N; if N is more than noise_frm_num, find the value at the range of noise_frm_num |
| 29 | RW | 0x0 | clean_noise_at_begin 1'b0: The noise will be clean only at the begin of the first time VAD is enable after reset 1'b1: The noise will be clean every time at the begin of VAD is enable |
| 28 | RW | 0x0 | force_noise_clk_en Force noise calculate clk enable 1'b0: The clock will be auto gating for low power 1'b1: The clock will be always enable |
| 27 | RO | 0x0 | Reserved |
| 26 | RW | 0x1 | clean_iir_en Clean IIR filter when VAD is disable 1'b0: Not clean 1'b1: Clean |
| 25:16 | RW | 0x0ff | noise_sample_num The number of sample in one frame to calculate the noise, the value N means N+1 sample. When this value is higher, the voice detect condition is more strict |
| 15:0 | RW | 0x0064 | sound_thd Initial sound threshold When this value is higher, the voice detect condition is more strict |

VAD DET CON2

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:16 | RW | 0x3bf5 | iir_anum_0 IIR numerator coefficient b0 |
| 15:8 | RW | 0xe6 | noise_alpha The update smooth speed of noise When this value is lower, the voice detect condition is more strict |
| 7 | RO | 0x0 | Reserved |
| 6:0 | RW | 0x63 | noise_frm_num The number of frame to calculate the noise, the value N means N+1 frame. When this value is lower, the voice detect condition is more strict |

VAD DET CON3

Address: Operational Base + offset (0x0068)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x3bf5 | iir_anum_2 IIR numerator coefficient b2 |
| 15:0 | RW | 0x8817 | iir_anum_1 IIR numerator coefficient b1 |

VAD DET CON4

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x382b | iir_aden_2 IIR demoninator coefficient a2 |
| 15:0 | RW | 0x8858 | iir_aden_1 IIR demoninator coefficient a1 |

VAD DET CON5

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0000 | iir_result Voice real time data after IIR filter |
| 15:0 | RW | 0x0000 | noise_abs Noise abs value |

VAD INT

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | Reserved |
| 11 | W1 C | 0x0 | vad_data_trans_int VAD data transfer interrupt 1'b0: Interrupt not generated 1'b1: Interrupt generated |
| 10 | RW | 0x0 | vad_data_trans_int_en VAD data transfer interrupt enable 1'b0: Disable 1'b1: Enable |
| 9 | RW | 0x0 | vad_idle VAD idle flag 1'b0: Not idle 1'b1: Idle |
| 8 | RO | 0x0 | ramp_loop_flag RAM adress loop flag 1'b0: Not loop 1'b1: Loop |

| Bit | Attr | Reset Value | Description |
|-----|---------|-------------|--|
| 7 | W1 C | 0x0 | work_timeout_int Work timeout interrupt 1'b0: Interrupt not generated 1'b1: Interrupt generated |
| 6 | W1 C | 0x0 | idle_timeout_int Idle timeout interrupt 1'b0: Interrupt not generated 1'b1: Interrupt generated |
| 5 | RW | 0x0 | error_int Error interrupt 1'b0: Interrupt not generated 1'b1: Interrupt generated |
| 4 | W1 C | 0x0 | vad_det_int VAD detect interrupt 1'b0: Interrupt not generated 1'b1: Interrupt generated |
| 3 | RW | 0x0 | work_timeout_int_en Work timeout interrupt enable 1'b0: Disable 1'b1: Enable |
| 2 | RW | 0x0 | idle_timeout_int_en Idle timeout interrupt enable 1'b0: Disable 1'b1: Enable |
| 1 | RW | 0x0 | error_int_en Error interrupt enable 1'b0: Disable 1'b1: Enable |
| 0 | RW | 0x0 | vad_det_int_en VAD detect interrupt enable 1'b0: Disable 1'b1: Enable |

VAD_AUX CONTROL

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:30 | RO | 0x0 | Reserved |
| 29 | RW | 0x0 | sample_cnt_en Sample counter enable 1'b0: Disable 1'b1: Enable |

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 28 | RW | 0x0 | int_trig_ctrl_en The VAD detection interrupt trigger control enable. 1'b0: Disable, the VAD detection interrupt is always triggered 1'b1: Enable, the VAD detection interrupt trigger is controlled by int_trig_valid_thd |
| 27:16 | RW | 0x000 | int_trig_valid_thd VAD detection interrupt trigger valid threshold. The VAD detection interrupt will be triggered valid after sample_cnt exceed int_trig_valid_thd. The value N means N+1, The unit is one voice sample point. |
| 15:12 | RO | 0x0 | Reserved |
| 11:4 | RW | 0x00 | data_trans_kbyte_thd Data transfer number threshold, the unit is KByte. The value N means N+1 KByte. The interrupt is generated per data_trans_kbyte_thd KBytes. |
| 3 | RO | 0x0 | Reserved |
| 2 | RW | 0x0 | data_trans_trig_int_en Trigger an interrupt when data transfer counter exceed data_trans_kbyte_thd every time. The data transfer counter will be cleared to 0 when the interrupt is asserted. It's valid only when bus_write_en=1. 1'b0: Disable 1'b1: Enable |
| 1 | RW | 0x0 | dis_ram_itf Disable write voice data to Internal SRAM through RAM interface 1'b0: Enable ram interface 1'b1: Disable ram interface |
| 0 | RW | 0x0 | bus_write_en Enable write voice data to Internal SRAM through AHB bus interface 1'b0: Disable 1'b1: Enable |

VAD SAMPLE CNT

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|------------------------------|
| 31:0 | RO | 0x00000000 | sample_cnt Sample counter |

VAD NOISE DATA

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RW | 0x0000 | noise_data Noise data |

11.5 Application Notes

11.5.1 VAD usage flow

VAD usage flow is as following figure.

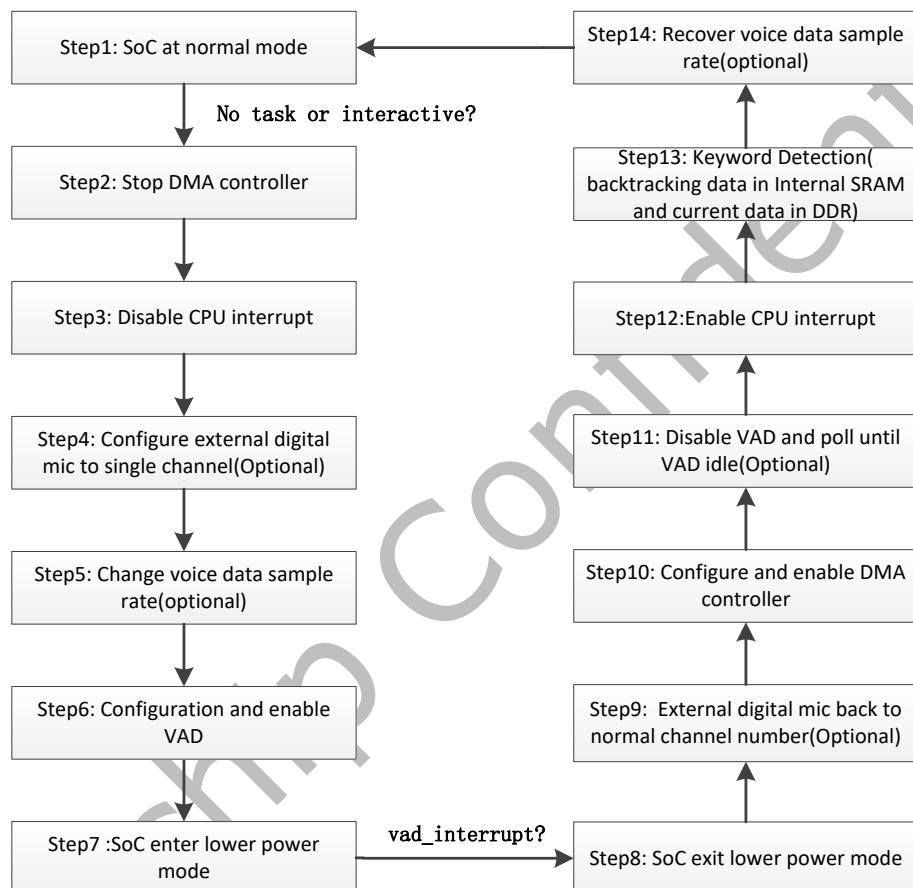


Fig. 11-3 VAD usage flow

- Step4, step5, step9, step14 are optional, user should consider the power consumption and keyword detection accuracy for these steps.
- Step11 is optional, user can keep VAD working, and use it as a DMA Controller. In this case keyword detection will used all data in internal SRAM.
- For step4, just power down the unused ADC in unused external digital mic. The I2S_8CH /2CH or PDM is still work at normal channel, the voice data of unused channels are zero.
- At Step9, power up ADC in external digital mic which is power down at step4. It's controlled by VAD or hardware after voice detection event, CPU needn't to configure it.

11.5.2 VAD configuration usage flow

1. Set VAD_VS_ADDR.vs_addr=I2S0_8CH base address + I2S_8CH_INCR_RXDR.
2. Set VAD_RAM_BEGIN_ADDR.ram_begin_addr and VAD_RAM_END_ADDR.ram_end_addr, the address should match with the Internal SRAM sharing scheme.
Set VAD_AUX_CONTROL.bus_write_en to enable write data through bus.

3. Adjust the sensitivity of voice activity detect by setting follow registers:

VAD_DET_CON0.noise_level
VAD_DET_CON0.vad_con_thd
VAD_DET_CON0.dis_vad_con_thd
VAD_DET_CON0.vad_thd_mode
VAD_DET_CON1.noise_sample_num
VAD_DET_CON1.sound_thd
VAD_DET_CON2.noise_frm_num
VAD_DET_CON2.noise_alpha

4. Set the iir_anum_0~3 and iir_aden_1~2 to adjust the IIR HPF coefficient.

For 48Khz sample rate:

iir_anum_0: 0x382d
iir_anum_1: 0x8fa5
iir_anum_2: 0x382d
iir_aden_1: 0x909b
iir_aden_2: 0x3150

For 16Khz sample rate (default):

iir_anum_0: 0x3bf5
iir_anum_1: 0x8817
iir_anum_2: 0x3bf5
iir_aden_1: 0x8858
iir_aden_2: 0x382b

For 8Khz sample rate:

iir_anum_0: 0x3e9f
iir_anum_1: 0x82c2
iir_anum_2: 0x3e9f
iir_aden_1: 0x82c9
iir_aden_2: 0x3d46

5. Set DET_CON5.noise_abs to ambient noise which is calculated by software.

Set VAD_NOISE_DATA+offset to initial the noise data of all frames. The frame number is VAD_DET_CON2.noise_frm_num. The first frame noise data address is VAD_NOISE_DATA, the second frame noise data address is VAD_NOISE_DATA+0x4, and so on.

6. Set VAD_INT.vad_det_int_en=0x1 to enable the interrupt.

7. Set VAD_AUX_CONTROL to disable detection at the beginning after VAD is enabled.

Set sample_cnt_en=0x1

Set int_trig_ctrl_en=0x1

Set int_trig_valid_thd to appropriate value, it recommended configuration to 4ms.

For 48Khz sample rate: int_trig_valid_thd= 0xc0

For 16Khz sample rate: int_trig_valid_thd= 0x40

For 8Khz sample rate: int_trig_valid_thd= 0x20

8. Set VAD_CONTROL register:

Set source_select=0x0, select I2S0_8CH

Set source_burst=0x3, select INCR4 burst type

Set source_burst_num=0x0, select 1 burst transfer per DMA request

Set vad_mode=0x0, select Mode 0

Set voice_channel_num=0x7, all voice channel number is 8

Set voice_channel_bitwidth=0x0, voice data width is 16 bits

Set vad_det_channel=0x0, use channel 0 to voice activity detect

Set vad_en=0x1, enable VAD

10. After above setting, VAD will start to work and system can enter low power mode.

11.5.3 Data Transfer Interrupt usage flow

When VAD is working at normal mode. User can get data transfer interrupt by following additional configuration.

1. Set VAD_AUX_CONTROL.data_trans_trig_int_en=0x1

2. Set VAD_AUX_CONTROL.data_trans_kbyte_thd at appropriate vaule

3. Set VAD_INT.vad_data_trans_int_en=0x1

11.5.4 Timeout configuration usage flow

1. Set VAD_TIMEOUT.idle_timeout_thd=0xffff, set VAD_TIMEOUT.idle_timeout_en=0x1, set VAD_INT.idle_timeout_int=0x1. After above setting, a counter is increase at AHB clock when dmac_engine is idle, the counter will be clear to 0 once dmac_engine start to read voice data. An interrupt will be asserted when the counter up to idle_timeout_thd. This idle timeout is used for I2S/PDM work fail(Don't assert DMA request for a long time).
2. Set VAD_TIMEOUT.work_timeout_thd=0x3ff, set VAD_TIMEOUT.work_timeout_en=0x1, set VAD_INT.work_timeout_int=0x1. After above setting, a counter is increase at AHB clock when dmac_engine is busy, the counter will be clear to 0 once dmac_engine is idle. An interrupt will be asserted when the counter up to work_timeout_thd. This work timeout is used for bus transmission congestion(a burst transferring is not completed for a long time).

Chapter 12 MIPI DSI Host Controller

12.1 Overview

The Display Serial Interface (DSI) is part of a group of communication protocols defined by the MIPI Alliance. The MIPI DSI HOST Controller is a digital core that implements all protocol functions defined in the MIPI DSI Specification. The MIPI DSI HOST Controller provides an interface between the system and the MIPI DPHY, allowing the communication with a DSI-compliant display. The MIPI DSI HOST Controller supports one to four lanes for data transmission with MIPI DPHY.

The MIPI DSI HOST Controller supports the following features:

- Compliant with MIPI Alliance standards
- Support the DPI interface color coding mappings into 24-bit Interface
 - 16 bits per pixel, configurations 1,2, and 3
 - 18 bits per pixel, configurations 1 and 2
 - 24 bits per pixel
- Programmable polarity of all DPI interface signals
- Extended resolutions beyond the DPI standard maximum resolution of 800x480 pixels:
 - Up to 2047 vertical active lines
 - Up to 63 vertical back porch lines
 - Up to 63 vertical front porch lines
 - Maximum resolution is limited by available DSI Physical link bandwidth which depends on the number of lanes and maximum speed per lane
- Interface with MIPI DPHY following PHY Protocol Interface (PPI), as defined in MIPI Alliance Specification for DPHY
- Up to four DPHY Data Lanes
- Bidirectional communication and escape mode support through data lane 0
- Transmission of all generic commands
- ECC and Checksum capabilities
- End of Transmission Packet(EOTP)
- Ultra Low-Power mode
- Fault recovery schemes

12.2 Block Diagram

The following diagram shows the MIPI DSI HOST Controller architecture.

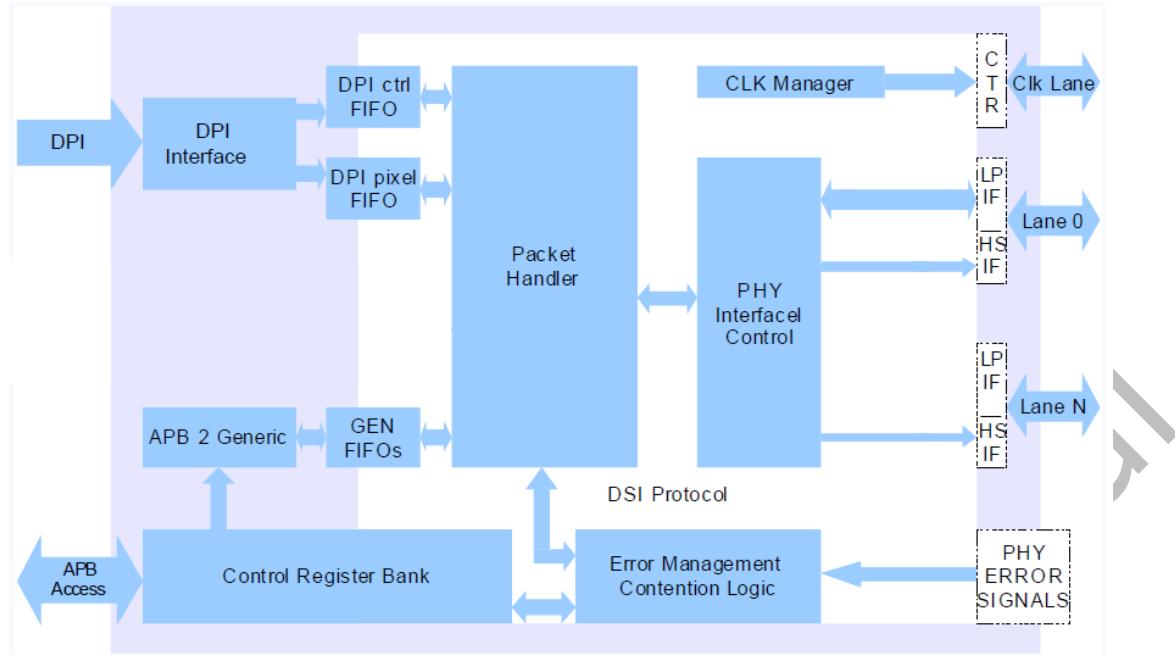


Fig. 12-1 MIPI DSI HOST Controller architecture

The DPI interface captures the data and control signals and conveys them to a FIFO for video control signals and another one for pixel data. This data is then used to build Video packets, hen in Video mode.

The Register Bank is accessible through a standard AMBA-APB slave interface, providing access to the MIPI DSI HOST Controller registers for configuration and control. There is also a fully programmable interrupt generator to inform the system about certain events.

The PHY Interface Control is responsible for managing the DPHY PPI interface. It acknowledges the current operation and enables low-power transmission/reception or a high-speed transmission. It also performs data splitting between available DPHY lanes for high-speed transmission.

The Packet Handler schedules the activities inside the link. It performs several functions based on the interfaces that are currently DPI and the video transmission mode that is used (burst mode or non-burst mode with sync pulse or sync events). It builds long or short packet generating correspondent ECC and CRC codes. This block also performs the following functions: Packet reception, Validation of packet header by checking the ECC, Header correction and notification for single-bit errors, Termination of reception, Multiple header error notification.

The APB-to-Generic block bridges the APB operations into FIFOs holding the Generic commands. The block interfaces with the following FIFOs: Command FIFO, Write payload FIFO, Read payload FIFO.

The Error Management notifies and monitors the error conditions on the DSI link. It controls the timers used to determine if a timeout condition occurred, performing an internal soft reset and triggering an interruption notification.

12.3 Function Description

12.3.1 DPI interface function

The DPI interface follows the MIPI DPI specification with pixel data bus width up to 24 bits. It is used to transmit the information in Video mode in which the transfers from the host processor to the peripheral take the form of a real-time pixel stream. This interface allows sending ShutDown (SD) and ColorMode (CM) commands, which are triggered directly by writing to the register of CFG_MISC_CON[2:1]. To transfer additional commands(for example, to initialize the display), use another interface such as APB Slave Generic Interface to complement the DPI interface.

The DPI interface captures the data and control signals and conveys them to the FIFO interfaces that transmit them to the DSI link. Two different streams of data are presented at the interface; video control signals and pixel data. Depending on the interface color

coding, the pixel data is disposed differently throughout the dpipixdata bus. The following table shows the Interface pixel color coding.

Table 12-1 Color table

| Signal Line | 16-bit | | | 18-bit | | 24-bit |
|--------------|----------|----------|----------|----------|----------|--------|
| | Config1 | Config2 | Config3 | Config1 | Config2 | |
| dpipixdata23 | Not used | R7 |
| dpipixdata22 | Not used | R6 |
| dpipixdata21 | Not used | Not used | R4 | Not used | R5 | R5 |
| dpipixdata20 | Not used | R4 | R3 | Not used | R4 | R4 |
| dpipixdata19 | Not used | R3 | R2 | Not used | R3 | R3 |
| dpipixdata18 | Not used | R2 | R1 | Not used | R2 | R2 |
| dpipixdata17 | Not used | R1 | R0 | R5 | R1 | R1 |
| dpipixdata16 | Not used | R0 | Not used | R4 | R0 | R0 |
| dpipixdata15 | R4 | Not used | Not used | R3 | Not used | G7 |
| dpipixdata14 | R3 | Not used | Not used | R2 | Not used | G6 |
| dpipixdata13 | R2 | G5 | G5 | R1 | G5 | G5 |
| dpipixdata12 | R1 | G4 | G4 | R0 | G4 | G4 |
| dpipixdata11 | R0 | G3 | G3 | G5 | G3 | G3 |
| dpipixdata10 | G5 | G2 | G2 | G4 | G2 | G2 |
| dpipixdata9 | G4 | G1 | G1 | G3 | G1 | G1 |
| dpipixdata8 | G3 | G0 | G0 | G2 | G0 | G0 |
| dpipixdata7 | G2 | Not used | Not used | G1 | Not used | B7 |
| dpipixdata6 | G1 | Not used | Not used | G0 | Not used | B6 |
| dpipixdata5 | G0 | Not used | B5 | B5 | B5 | B5 |
| dpipixdata4 | B4 | B4 | B4 | B4 | B4 | B4 |
| dpipixdata3 | B3 | B3 | B3 | B3 | B3 | B3 |
| dpipixdata2 | B2 | B2 | B2 | B2 | B2 | B2 |
| dpipixdata1 | B1 | B1 | B1 | B1 | B1 | B1 |
| dpipixdata0 | B0 | B0 | Not used | B0 | B0 | B0 |

The DPI interface can be configured to increase flexibility and promote correct usage of this interface for several systems. These configuration options are as follows: Polarity control: All the control signals are programmable to change the polarity depending on system requirements.

After the MIPI DSI HOST Controller reset, DPI waits for the first VSYNC active transition to start signal sampling, including pixel data, and preventing image transmission in the middle of a frame.

If interface pixel color coding is 18 bits and the 18-bit loosely packed stream is disabled, the number of lines programmed in the pixels per lines configuration is a multiple of four. This means that in this mode, the two LSBs in the configuration are always inferred as zero. The specification states that in this mode, the pixel line size should be a multiple of four.

12.3.2 APB Slave Generic Interface

The APB Slave interface allows the transmission of generic information in Command mode, and follows the proprietary register interface. Commands sent through this interface are not constrained to comply with the DCS specification, and can include generic commands described in the DSI specification as manufacturer-specific.

The MIPI DSI HOST Controller supports the transmission or write and read command mode packets as described in the DSI specification. These packets are built using the APB register access. The GEN_PLD_DATA register has two distinct functions based on the operation.

Writing to this register sends the data as payload when sending a Command mode packet. Reading this register returns the payload of a read back operation. The GEN_HDR register contains the Command mode packet header type and header data. Writing to this register

triggers the transmission of the packet implying that for a long Command mode packet, the packet's payload needs to be written in advance in the GEN_PLD_DATA register.

The valid packets available to be transmitted through the Generic interface are as follows:

- Generic Write Short Packet 0 Parameters
- Generic Write Short Packet 1 Parameters
- Generic Write Short Packet 2 Parameter
- Generic Write Short Packet 0 Parameter
- Generic Write Short Packet 1 Parameters
- Generic Write Short Packet 2 Parameter
- Maximum Read Packet Configuration
- Generic Long Write Packet
- DCS Write Short Packet 0 Parameter
- DCS Write Short Packet 1 Parameter
- DCS Write Short Packet 0 Parameter
- DCS Write Long Packet

A set of bits in the CMD_PKT_STATUS register report the status of the FIFOs associated with APB interface support.

Generic interface packets are always transported using one of the DSI transmission modes; Video mode or Command mode. If neither of these mode are selected, the packets are not transmitted through the link and the released FIFOs eventually get overflowed.

The transfer of packets through the APB bus is based on the following conditions:

The APB protocol defines that the write and read procedure takes two clock cycles each to be executed. This means that the maximum input data rate through the APB interfaces is always half the speed of the APB clock.

The data input bus has a maximum width of 32 bits. This allows for a relation to be defined between the input APB clock frequency and maximum bi rate achievable by the APB interface.

The DSI link bit rate when using solely APB is equal to (APB clock frequency) *16 Mbps.

The bandwidth is dependent on the APB clock frequency; the available bandwidth increases with the clock frequency.

To drive the APB interface to achieve high bandwidth Command mode traffic transported by the DSI link, the MIPI DSI HOST Controller should operate in the Command mode only and the APB interface should be the only data source that is currently in use. Thus, the APB interface has the entire bandwidth of the DSI link and does not share it with any another input interface source.

The memory write commands require maximum throughout from the APB interface, because they contain the most amount of data conveyed by the DSI link. While writing the packet information, first write the payload of a given packet into the payload FIFO using the GEN_PLD_DATA register. When the payload data is for the command parameters, place the first byte to be transmitted in the least significant byte position of the APB data bus.

After writing the payload, write the packet header into the command FIFO. For more information and it should follow the pixel to byte conversion organization referred in the Annexure A of the DCS specification. The follow figures show how the pixel data should be organized in the APB data write bus. The memory write commands are conveyed in DCS long packets. DCS long packets are encapsulated in a DSI packet. The DSI included in the diagrams. In the follow figures, the Write Memory Command can be replaced by the DCS command Write Memory Start and Write Memory Continue.

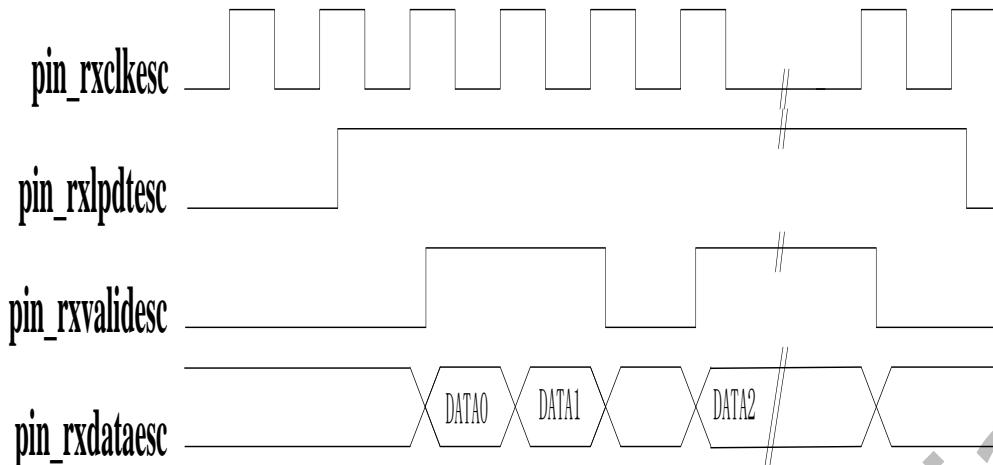


Fig. 12-2 24 bpp APB Pixel to Byte Organization

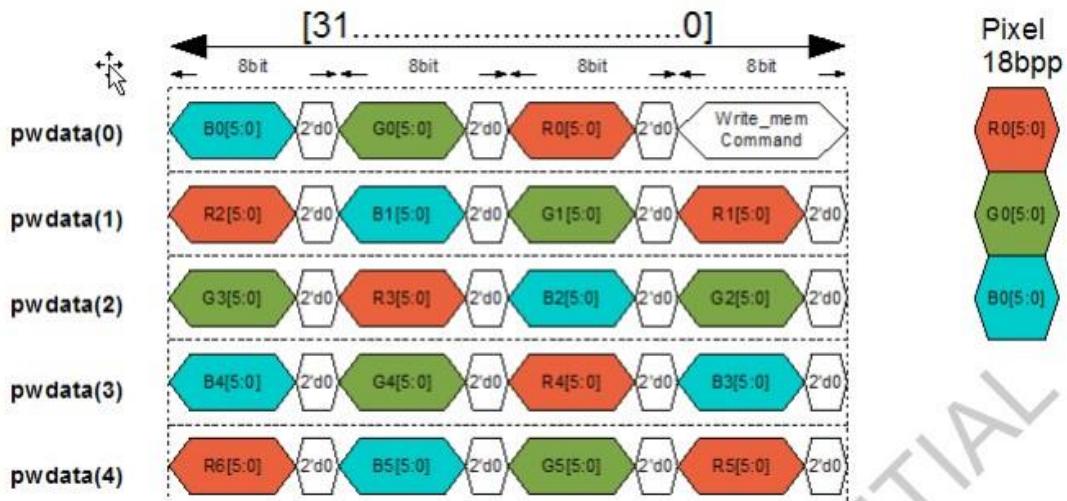


Fig. 12-3 18 bpp APB Pixel to Byte Organization

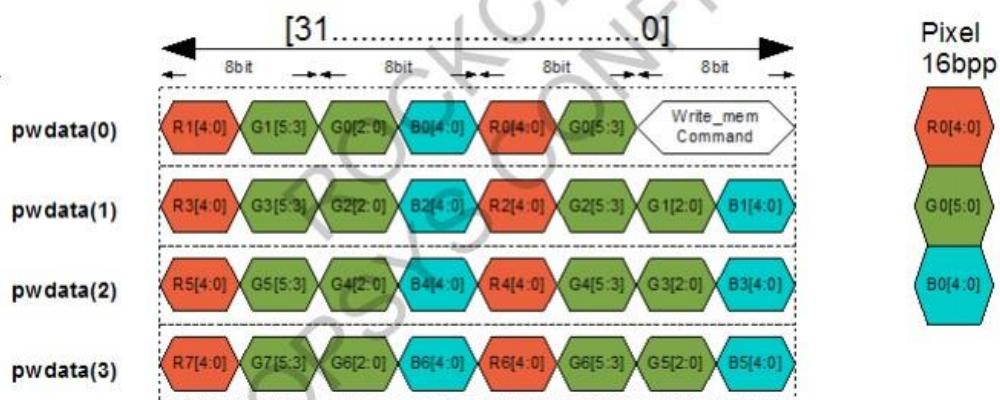


Fig. 12-4 16 bpp APB Pixel to Byte Organization

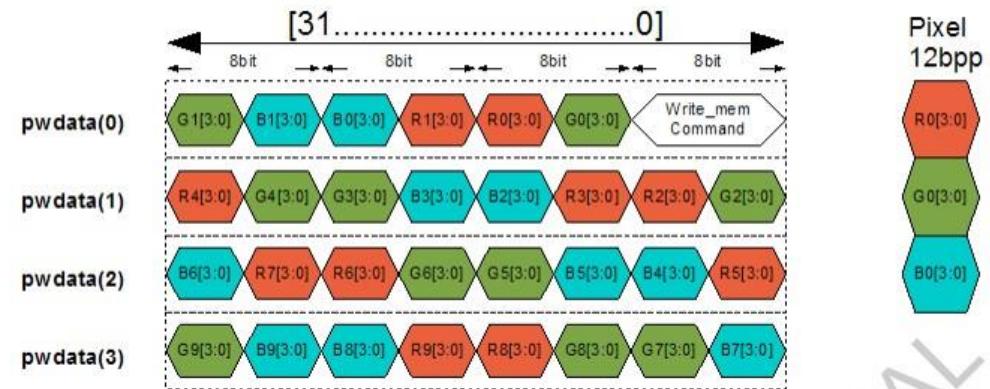


Fig. 12-5 12 bpp APB Pixel to Byte Organization

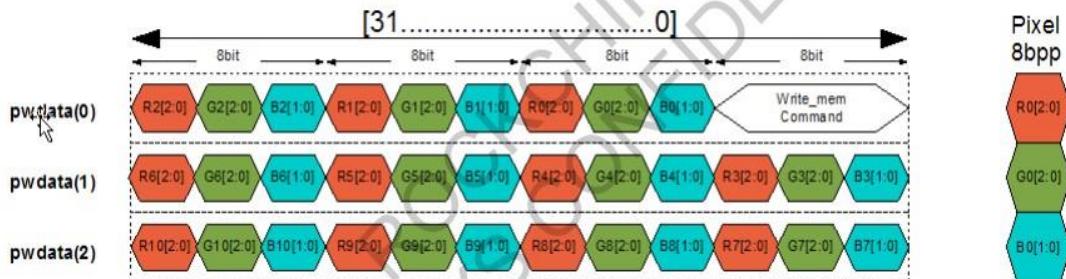


Fig. 12-6 8bpp APB Pixel to Byte Organization

12.3.3 Transmission of Commands in Video Mode

The MIPI DSI HOST Controller supports the transmission of commands, both in high-speed and low-power, while in Video mode. The DSI controller uses Blanking or Low-Power(BLLP) periods to transmit commands inserted through the APB Generic interface. Those periods correspond to the shaded areas of the following figure.

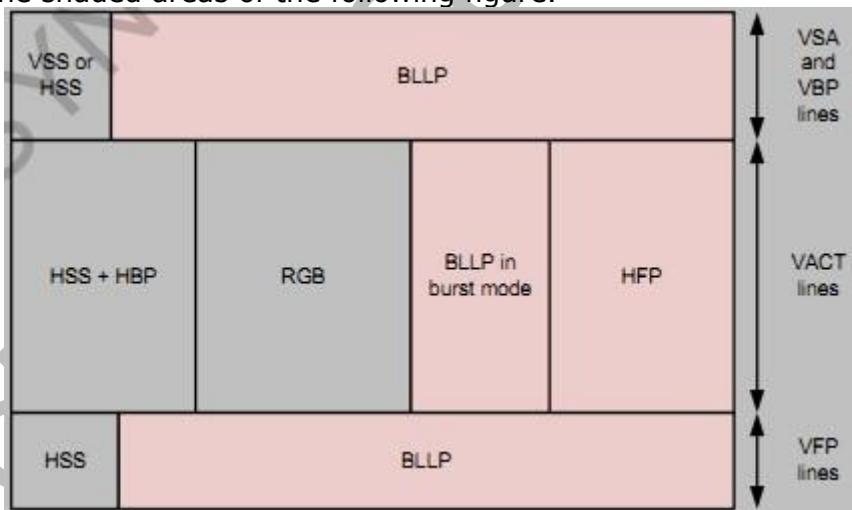


Fig. 12-7 Command Transmission Periods within the Image Area

Commands are transmitted in the blanking periods after the following packet/states:

- Vertical Sync Start (VSS) packets, if the Video Sync pulses are not enabled
- Horizontal Sync End (HSE) packets, in the VSA, VBP, and VFP regions
- Horizontal Sync Start (HSS) packets, if the Video Sync pulses are not enabled in the VSA, VBP, and VFP regions
- Horizontal Active (HACT) state

Only one command is transmitted per line, even in the case of the last line of a frame but one command is possible for each line.

The MIPI DSI HOST Controller avoids sending commands in the last line because it is possible that the last line is shorter than the other ones. For instance, the line time (tL) could be half a cycle longer than the tL on the DPI interface, that is, each line in the frame taking half a cycle from time for the last line. This results in the last line being $(1/2 \text{ cycle}) * (\text{number of lines} - 1)$ shorter than tL.

The dpicolorm and dpishutdn input signals are also able to trigger the sending of command packets. The commands are DSI data types Color Mode On, Color Mode Off, Shut Down Peripheral, and Turn on Peripheral. These commands are not sent in the VACT region. If the lpcmden bit of the VID_MODE_CFG register is 1, these commands are sent in LP mode. In LP mode, the ouvact_lpcmd_time field of the LP_CMD_TIM register is used to determine if these commands can be transmitted. It is assumed that outvact_lpcmd_time is greater than or equal to 4 bytes (number of bytes in a short packet), because the DSI HOST does not transmit these commands on the last line.

If the frame_BTA_ack field is set in the VID_MODE_CFG register, a BTA is generated by DSI HOST after the last line of a frame. This may coincide with a write command or a read command. In either case, the edpihalt signal is held asserted until an acknowledge has been received (control of the DSI bus is returned to the host).

If the lpcmden bit of the VID_MODE_CFG register is set to 1, the commands are sent in low-power in Video mode. In this case, it is necessary to calculate the time available, in bytes, to transmit a command in LP mode for Horizontal Front Porch (HFP), Vertical Sync Active (VSA), Vertical Back Porch (VBP), and Vertical Front Porch(VFP) regions.

The outvact_lpcmd_time field of the LP_CMD_TIM register indicates the time available (in bytes) to transmit a command in LP mode, based on the escape clock, on a line during the VSA, VBP, and the VFP

Outvact_lpcmd_time=(tL-(Time to transmit HSS and HSE frames +tHSA+Time to enter and leave LP mode+Time to send the DPHY LPDT command)) / escape clock period / 8 / 2

Where,

tL=Line time

tHSA=Time to send a short packet (for sync events) or time of the HAS pulse (for sync pulses)

In the above equation, division by eight is done to convert the time available to bytes and division by two is done because one bit is transmitted once in every two escape clock cycles.

The outvact_lpcmd_time filed can be compared directly with the size of the command to be transmitted to determine if there is enough time to transmit the command. The maximum size of a command that can be transmitted in LP mode is limited to 255 bytes by this field. This register must be programmed to a value greater than or equal to 4 bytes for the transmission of the DCTRL commands such as shutdown and colorm in LP mode.

Consider an example with 12.6 μ s per line and assume an escape clock frequency of 15 MHz. In this case, 189 escape clock cycles are available to enter and exit LP mode and transmit command. The following are assumed:

Sync pulses are not being transmitted

Two lane byte clock ticks are required to transmit a short packet

phy_lp2hs_time=16
phy_lp2p_time=20

In this example, a 11-byte command can be transmitted as follows:

outvact_lpcmd_time=(12.6 μ s-(2*10 ns)-(16*10 ns)-(20*10 ns)-(8*66 ns)) / 66 ns / 8 / 2
= 11 bytes

The invact_lpcmd_time field of the LP_CMD_TIM register indicates the time available (in bytes) to transmite a command in LP mode (based on the escape clock) in the Vertical Active (VACT) region. This time is calculated as follows:

Invact_lpcmd_time=((tHFP-Time to enter and leave low-power mode + Blanking period before the HFP when in Burst mode- Time to send the DPHY LPDT command) / escape clock period) / 8

Where,

tHFP=line time-tHSA-tHBP-tHACT

tHACT=vid_pkt_size*bits_per_pixel*lane_byte_clock_period / num_lanes

The invact_lpcmd_time field can be compared directly with the size of the command to be transmitted to determined if there is time to transmit the command.

Consider an example where the refresh rate is 60 Hz. The number of lines is 1320 (typical). The tL in this case is 12.6 μ s. With a lane byte clock of 100 MHz, 1260 clock ticks are availabel to transmit a single frame. If 800 ticks are used for pixel data then 460 ticks

(4.6 μ s) are available for Horizontal Sync Start (HSS), HFP, and HBP. Assuming that 2.3 μ s is available for HFP and the escape clock is 15MHz, only 34 LP clock ticks are available to enter LP, transmit a command, and return from LP mode. Approximately 12 escape clock ticks are required to enter and leave LP mode. Therefore, only 1 byte could be transmitted in this period.

A short packet (for example, generic short write) requires a minimum of 4 bytes. Therefore, in this example, commands are not sent in the VACT region. If Burst mode is enabled, more time is available to transmit commands in the VACT region. The following are assumed:

The controller is not in Burst mode

```
phy_lp2hs_time=16
phy_lp2hs_time=16
```

In this example invact_lpcmd_time is calculated as follows:

$$\text{Invact_lpcmd_time} = (2.3\mu\text{s} - (16*10\text{ ns}) - (20*10\text{ ns}) - (8*66\text{ ns})) / 66\text{ ns} / 8 = 2 \text{ bytes}$$

The outvact_lpcmd_time and invact_lpcmd_time fields allow a simple comparison to determine if a command can be transmitted in any of the BLLP periods.

Following figure illustrates the meaning of invact_lpcmd_time and outvact_lpcmd_time, matching them with the shaded areas and the VACT region.

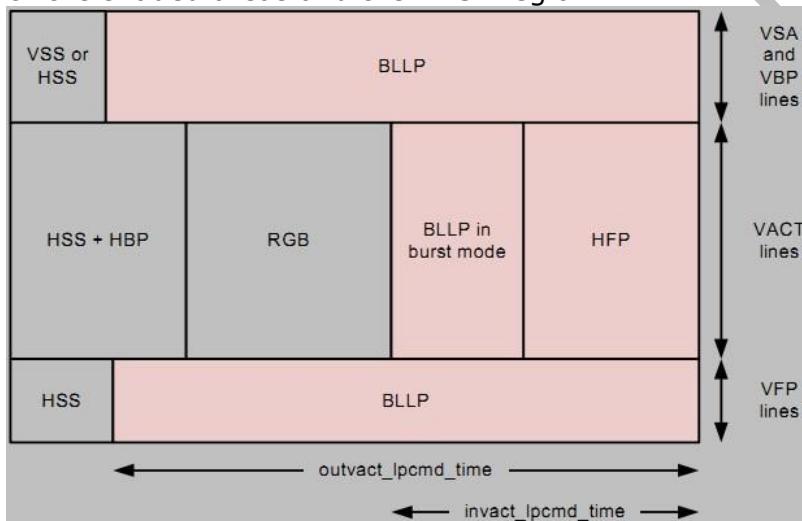


Fig. 12-8 Location in the Image Area

If the lpcmden bit of the VID_MODE_CFG register is 0, the commands are sent in high_speed in Video Mode. In this case, the DSI HOST automatically determines the area where each command can be sent and no programming or calculation is required.

On read command Transmission, the max_rd_time field of the PHY_TMR_CFG register configures the maximum amount of time required to perform a read command in lane byte clock cycles.

The maximum time required to perform a read command in Lane byte clock cycles (max_rd_time) = Time to transmit the read command in LP mode + Time to enter and leave LP mode + Time to return the read data packet from the peripheral device.

The time to return the read data packet from the peripheral depends on the number of bytes read and the escape clock frequency of the peripheral; not the escape clock of the host. The max_rd_time field is used in both HS and LP mode to determine if there is time to complete a read command in a BLLP period.

In high-speed mode (lpcmden=0), max_rd_time is calculated as follows:

max_rd_time = phy_hs2lp_time + Time to return the read data packet from the peripheral device + phy_hs2hs_time

In low-power mode (lpcmden = 1), max_rd_time is calculated as follows:

max_rd_time = phy_hs2lp_time + LPDT command time + Read command time in LP mode + Time to return the data read from the peripheral device + phy_lp2hs_time

Where,

LPDT command time = (8*Host escape clock period) / Lane byte clock period

Read command time in LP mode = (32 * host escape clock period) / lane byte clock period

It is recommended to keep the maximum number of bytes read from the peripheral to a

minimum to have sufficient time available to issue the read commands on a line. Ensure that max_rd_time* Lane byte clock period is less than outvact_lpcmd_time *8*Escape clock period of the host.

Otherwise, the read commands are serviced on the last line of a frame and the edpihalt signal may be asserted. If it is necessary to read a large number of parameters (>16), increase the max_rd_time while the read command is being executed. When the read has completed, decrease the max_rd_time to a lower value.

12.4 Register Description

12.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

12.4.2 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|---------------------------------|--------|------|-------------|---|
| MIPI_DSI_HOST_VERSIO_N | 0x0000 | W | 0x3133302a | Version |
| MIPI_DSI_HOST_PWR_UP | 0x0004 | W | 0x00000000 | Power up |
| MIPI_DSI_HOST_CLKMGR_CFG | 0x0008 | W | 0x00000000 | Configure the factor of the internal dividers to divide lanebyteclk |
| MIPI_DSI_HOST_DPI_VCID | 0x000c | W | 0x00000000 | DPI virtual channel id |
| MIPI_DSI_HOST_DPI_COL_OR_CODING | 0x0010 | W | 0x00000000 | DPI color coding |
| MIPI_DSI_HOST_DPI_CFG_POL | 0x0014 | W | 0x00000000 | Polarity of the DPI signals |
| MIPI_DSI_HOST_DPI_LP_CMD_TIM | 0x0018 | W | 0x00000000 | The timing for the low-power command while in video mode |
| MIPI_DSI_HOST_PCKHDL_CFG | 0x002c | W | 0x00000000 | Configures how EoTp,BTA,CRC and ECC are to be used , to meet the peripheral's characteristics |
| MIPI_DSI_HOST_GEN_VCID | 0x0030 | W | 0x00000000 | The virtual channel id of the read responses to store and return to the Generic interface |
| MIPI_DSI_HOST_MODE_CFG | 0x0034 | W | 0x00000001 | This register configures the mode of operation - Video mode or Command mode(Command can be sent even in video mode too) |
| MIPI_DSI_HOST_VID_MODE_CFG | 0x0038 | W | 0x00000000 | Configure several aspects of the Video mode operation |
| MIPI_DSI_HOST_VID_PKT_SIZE | 0x003c | W | 0x00000000 | The video packet size |
| MIPI_DSI_HOST_VID_NUM_CHUNKS | 0x0040 | W | 0x00000000 | The number of chunks to use |
| MIPI_DSI_HOST_VID_NULL_SIZE | 0x0044 | W | 0x00000000 | The size of null packets |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------|--------|------|-------------|---|
| MIPI_DSI_HOST_VID_HSA_TIME | 0x0048 | W | 0x00000000 | HSA time |
| MIPI_DSI_HOST_VID_HBP_TIME | 0x004c | W | 0x00000000 | HBP time |
| MIPI_DSI_HOST_VID_HLINE_TIME | 0x0050 | W | 0x00000000 | The overall time for video line |
| MIPI_DSI_HOST_VID_VSA_LINES | 0x0054 | W | 0x00000000 | VSA period |
| MIPI_DSI_HOST_VID_VBP_LINES | 0x0058 | W | 0x00000000 | VBP period |
| MIPI_DSI_HOST_VID_VFP_LINES | 0x005c | W | 0x00000000 | VFP period |
| MIPI_DSI_HOST_VID_VACTIVE_LINES | 0x0060 | W | 0x00000000 | The vertical resolution of the video |
| MIPI_DSI_HOST_EDPI_CMD_SIZE | 0x0064 | W | 0x00000000 | The size of eDPI packet |
| MIPI_DSI_HOST_CMD_MODE_CFG | 0x0068 | W | 0x00000000 | Configure several aspects of the command mode operation |
| MIPI_DSI_HOST_GEN_HDR | 0x006c | W | 0x00000000 | Set the header for new packets sent using the Generic interface |
| MIPI_DSI_HOST_GEN_PLD_DATA | 0x0070 | W | 0x00000000 | The payload for the packet sent using the Generic interface |
| MIPI_DSI_HOST_CMD_PKT_STATUS | 0x0074 | W | 0x00000000 | The status of FIFOs related to the Generic interface |
| MIPI_DSI_HOST_TO_CNT_CFG | 0x0078 | W | 0x00000000 | The counters that trigger the timeout error |
| MIPI_DSI_HOST_HS_RD_TO_CNT | 0x007c | W | 0x00000000 | The Peripheral Response timeout after high-speed read operation |
| MIPI_DSI_HOST_LP_RD_TO_CNT | 0x0080 | W | 0x00000000 | The Peripheral Response timeout after low-power read operation |
| MIPI_DSI_HOST_HS_WR_TO_CNT | 0x0084 | W | 0x00000000 | The Peripheral Response timeout after high-speed write operation |
| MIPI_DSI_HOST_LP_WR_TO_CNT | 0x0088 | W | 0x00000000 | The Peripheral Response timeout after low-power write operation |
| MIPI_DSI_HOST_BTA_TO_CNT | 0x008c | W | 0x00000000 | The Peripheral Response timeout after the Bus Turnaround completion |
| MIPI_DSI_HOST_SDF_3D | 0x0090 | W | 0x00000000 | The 3D control information for VSS Packets in video mode |
| MIPI_DSI_HOST_LPCLK_CTRL | 0x0094 | W | 0x00000000 | Using non-continuous clock in the clock lane |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------------|--------|------|-------------|--|
| MIPI_DSI_HOST_PHY_TMR_LPCLK_CFG | 0x0098 | W | 0x00000000 | Set the time that the HOST assumes in calculations for the clock lane to switch between high-speed and low-power |
| MIPI_DSI_HOST_PHY_TMR_CFG | 0x009c | W | 0x00000000 | Set the time that the HOST assumes in calculations for the data lanes to switch between high-speed and low-power |
| MIPI_DSI_HOST_PHY_RSTZ | 0x00a0 | W | 0x00000000 | The reset of the PLL and DPHY |
| MIPI_DSI_HOST_PHY_IF_CFG | 0x00a4 | W | 0x00000003 | The minimum time to remain in Stop state |
| MIPI_DSI_HOST_PHY_STATUS | 0x00b0 | W | 0x00000000 | The DPHY status |
| MIPI_DSI_HOST_INT_ST0 | 0x00bc | W | 0x00000000 | The status of the interrupt sources |
| MIPI_DSI_HOST_INT_ST1 | 0x00c0 | W | 0x00000000 | The status of the interrupt sources |
| MIPI_DSI_HOST_INT_MSK0 | 0x00c4 | W | 0x00000000 | The masks of interrupt sources |
| MIPI_DSI_HOST_INT_MSK1 | 0x00c8 | W | 0x00000000 | The masks of interrupt sources |
| MIPI_DSI_HOST_INT_FORCE0 | 0x00d8 | W | 0x00000000 | Force interrupts |
| MIPI_DSI_HOST_INT_FORCE1 | 0x00dc | W | 0x00000000 | Force interrupts |
| MIPI_DSI_HOST_VID_SHADOW_CTRL | 0x0100 | W | 0x00000000 | Control the DPI shadow register |
| MIPI_DSI_HOST_DPI_VCID_ACT | 0x010c | W | 0x00000000 | The virtual channel id for the DPI traffic |
| MIPI_DSI_HOST_DPI_COLORMOD_CODING_ACT | 0x0110 | W | 0x00000000 | DPI color coding |
| MIPI_DSI_HOST_DPI_LPCMD_TIM_ACT | 0x0118 | W | 0x00000000 | The timing for low-power commands sent while in Video mode |
| MIPI_DSI_HOST_VID_MODE_CFG_ACT | 0x0138 | W | 0x00000000 | Configure several aspects of Video mode operation |
| MIPI_DSI_HOST_VID_PKT_SIZE_ACT | 0x013c | W | 0x00000000 | Configure the video packet size |
| MIPI_DSI_HOST_VID_NUM_CHUNKS_ACT | 0x0140 | W | 0x00000000 | Configure the number of chunks to use |
| MIPI_DSI_HOST_VID_NULL_SIZE_ACT | 0x0144 | W | 0x00000000 | The size of null packets |

| Name | Offset | Size | Reset Value | Description |
|--------------------------------------|--------|------|-------------|--|
| MIPI DSI HOST VID HSA TIME ACT | 0x0148 | W | 0x00000000 | HSA time |
| MIPI DSI HOST VID HBP TIME ACT | 0x014c | W | 0x00000000 | HBP time |
| MIPI DSI HOST VID HLINE TIME ACT | 0x0150 | W | 0x00000000 | The overall time for each video line |
| MIPI DSI HOST VID VSA LINES ACT | 0x0154 | W | 0x00000000 | VSA period |
| MIPI DSI HOST VID VBP LINES ACT | 0x0158 | W | 0x00000000 | VBP period |
| MIPI DSI HOST VID VFP LINES ACT | 0x015c | W | 0x00000000 | VFP period |
| MIPI DSI HOST VID VA CTIVE LINES ACT | 0x0160 | W | 0x00000000 | The vertical resolution of video |
| MIPI DSI HOST SDF 3D ACT | 0x0190 | W | 0x00000000 | Store 3D control information for VSS in video mode |

Notes: **S**-ize: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

12.4.3 Detail Register Description

MIPI DSI HOST VERSION

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|-------------|
| 31:0 | RW | 0x3133302a | version |

MIPI DSI HOST PWR UP

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | shutdownz This bit configures the core either to power up or to reset. Its default value is 0. After the core configuration, to enable the DWC_mipi_dsi_host, set this register to 1. 1'b0: Reset 1'b1: Power-up |

MIPI DSI HOST CLKMGR CFG

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | <p>to_clk_division This field indicates the division factor for the Time Out clock used as the timing unit in the configuration of HS to LP and LP to HS transition error</p> |
| 7:0 | RW | 0x00 | <p>tx_esc_clk_division This field indicates the division factor for the TX Escape clock source (lanebyteclk). The values 0 and 1 stop the TX_ESC clock generation</p> |

MIPI DSI HOST DPI VCID

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | Reserved |
| 1:0 | RW | 0x0 | <p>dpi_vcidx This field configures the DPI virtual channel id that is indexed to the Video mode packets</p> |

MIPI DSI HOST DPI COLOR CODING

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:9 | RO | 0x0 | Reserved |
| 8 | RW | 0x0 | <p>loosely18_en When set to 1, this bit activates loosely packed variant to 18-bit configurations</p> |
| 7:4 | RO | 0x0 | Reserved |
| 3:0 | RW | 0x0 | <p>dpi_color_coding This field configures the DPI color coding as follows: 4'b0000: 16-bit configuration 1 4'b0001: 16-bit configuration 2 4'b0010: 16-bit configuration 3 4'b0011: 18-bit configuration 1 4'b0100: 18-bit configuration 2 4'b0101: 24-bit 4'b0110: 20-bit YCbCr 4:2:2 loosely packed 4'b0111: 24-bit YCbCr 4:2:2 4'b1000: 16-bit YCbCr 4:2:2 4'b1001: 30-bit 4'b1010: 36-bit 4'b1011-4'b1111: 12-bit YCbCr 4:2:0</p> |

MIPI DSI HOST DPI CFG POL

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | Reserved |
| 4 | RW | 0x0 | colorm_active_low When set to 1, this bit configures the color mode pin (dpicolorm) as active low |
| 3 | RW | 0x0 | shutd_active_low When set to 1, this bit configures the shutdown pin (dpishutdn) as active low |
| 2 | RW | 0x0 | hsync_active_low When set to 1, this bit configures the horizontal synchronism pin (dipihsync) as active low |
| 1 | RW | 0x0 | vsync_active_low When set to 1, this bit configures the vertical synchronism pin (dipivsync) as active low |
| 0 | RW | 0x0 | dataen_active_low When set to 1, this bit configures the data enable pin (dpidataen) as active low |

MIPI DSI HOST DPI LP CMD TIM

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | Reserved |
| 23:16 | RW | 0x00 | outvact_lpcmd_time This field is used for the transmission of commands in low-power mode. It defines the size, in bytes, of the largest packet that can fit in a line during the VSA, VBP, and VFP regions |
| 15:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0x00 | invact_lpcmd_time This field is used for the transmission of commands in low-power mode. It defines the size, in bytes, of the largest packet that can fit in a line during the VACT region |

MIPI DSI HOST PCKHDL CFG

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | Reserved |
| 4 | RW | 0x0 | crc_rx_en When set to 1, this bit enables the CRC reception and error reporting |
| 3 | RW | 0x0 | ecc_rx_en When set to 1, this bit enables the ECC reception, error correction, and reporting |
| 2 | RW | 0x0 | bta_en When set to 1, this bit enables the Bus Turn-Around (BTA) request |
| 1 | RW | 0x0 | eotp_rx_en When set to 1, this bit enables the EoTp reception |
| 0 | RW | 0x0 | eotp_tx_en When set to 1, this bit enables the EoTp transmission |

MIPI DSI HOST GEN VCID

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | Reserved |
| 1:0 | RW | 0x0 | gen_vcida_rx This field indicates the Generic interface read-back virtual channel identification |

MIPI DSI HOST MODE CFG

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x1 | cmd_video_mode This bit configures the operation mode: 1'b0: Video mode 1'b1: Command mode |

MIPI DSI HOST VID MODE CFG

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | Reserved |
| 24 | RW | 0x0 | vpg_orientation This field indicates the color bar orientation as follows: 1'b0: Vertical mode 1'b1: Horizontal mode |
| 23:21 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 20 | RW | 0x0 | vpg_mode This field is to select the pattern: 1'b0: Color bar (horizontal or vertical) 1'b1: BER pattern (vertical only) |
| 19:17 | RO | 0x0 | Reserved |
| 16 | RW | 0x0 | vpg_en When set to 1, this bit enables the video mode pattern generator |
| 15 | RW | 0x0 | lp_cmd_en When set to 1, this bit enables the command transmission only in lowpower mode |
| 14 | RW | 0x0 | frame_bta_ack_en When set to 1, this bit enables the request for an acknowledge response at the end of a frame |
| 13 | RW | 0x0 | lp_hfp_en When set to 1, this bit enables the return to low-power inside the Horizontal Front Porch (HFP) period when timing allows |
| 12 | RW | 0x0 | lp_hbp_en When set to 1, this bit enables the return to low-power inside the Horizontal Back Porch (HBP) period when timing allows |
| 11 | RW | 0x0 | lp_vact_en When set to 1, this bit enables the return to low-power inside the Vertical Active (VACT) period when timing allows |
| 10 | RW | 0x0 | lp_vfp_en When set to 1, this bit enables the return to low-power inside the Vertical Front Porch (VFP) period when timing allows |
| 9 | RW | 0x0 | lp_vbp_en When set to 1, this bit enables the return to low-power inside the Vertical Back Porch (VBP) period when timing allows |
| 8 | RW | 0x0 | lp_vsa_en When set to 1, this bit enables the return to low-power inside the Vertical Sync Time (VSA) period when timing allows |
| 7:2 | RO | 0x0 | Reserved |
| 1:0 | RW | 0x0 | vid_mode_type This field indicates the video mode transmission type as follows: 2'b00: Non-burst with sync pulses 2'b01: Non-burst with sync events 2'b10 and 2'b11: Burst mode |

MIPI DSI HOST VID PKT SIZE

Address: Operational Base + offset (0x003c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0000 | vid_pkt_size This field configures the number of pixels in a single video packet. For 18-bit not loosely packed data types, this number must be a multiple of 4. For YCbCr data types, it must be a multiple of 2, as described in the DSI specification |

MIPI DSI HOST VID NUM CHunks

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x0000 | vid_num_chunks This register configures the number of chunks to be transmitted during a Line period (a chunk consists of a video packet and a null packet). If set to 0 or 1, the video line is transmitted in a single packet. If set to 1, the packet is part of a chunk, so a null packet follows it if vid_null_size > 0. Otherwise, multiple chunks are used to transmit each video line |

MIPI DSI HOST VID NULL SIZE

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x0000 | vid_null_size This register configures the number of bytes inside a null packet. Setting it to 0 disables the null packets |

MIPI DSI HOST VID HSA TIME

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | Reserved |
| 11:0 | RW | 0x000 | vid_hsa_time This field configures the Horizontal Synchronism Active period in lane byte clock cycles |

MIPI DSI HOST VID HBP TIME

Address: Operational Base + offset (0x004c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | Reserved |
| 11:0 | RW | 0x000 | vid_hsa_time This field configures the Horizontal Synchronism Active period in lane byte clock cycles |

MIPI DSI HOST VID HLINE TIME

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:15 | RO | 0x0 | Reserved |
| 14:0 | RW | 0x0000 | vid_hline_time This field configures the size of the total line time (HSA+HBP+HACT+HFP) counted in lane byte clock cycles |

MIPI DSI HOST VID VSA LINES

Address: Operational Base + offset (0x0054)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | Reserved |
| 9:0 | RW | 0x000 | vsa_lines This field configures the Vertical Synchronism Active period measured in number of horizontal lines |

MIPI DSI HOST VID VBP LINES

Address: Operational Base + offset (0x0058)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | Reserved |
| 9:0 | RW | 0x000 | vbp_lines This field configures the Vertical Back Porch period measured in number of horizontal lines |

MIPI DSI HOST VID VFP LINES

Address: Operational Base + offset (0x005c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | vfp_lines This field configures the Vertical Front Porch period measured in number of horizontal lines |

MIPI DSI HOST VID VACTIVE LINES

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:14 | RO | 0x0 | Reserved |
| 13:0 | RW | 0x0000 | v_active_lines This field configures the Vertical Active period measured in number of horizontal lines |

MIPI DSI HOST EDPI CMD SIZE

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:16 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 15:0 | RW | 0x0000 | edpi_allowed_cmd_size This field configures the maximum allowed size for an eDPI write memory command, measured in pixels. Automatic partitioning of data obtained from eDPI is permanently enabled |

MIPI DSI HOST CMD MODE CFG

Address: Operational Base + offset (0x0068)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:25 | RO | 0x0 | Reserved |
| 24 | RW | 0x0 | max_rd_pkt_size This bit configures the maximum read packet size command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 23:20 | RO | 0x0 | Reserved |
| 19 | RW | 0x0 | dcs_lw_tx This bit configures the DCS long write packet command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 18 | RW | 0x0 | dcs_sr_0p_tx This bit configures the DCS short read packet with zero parameter command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 17 | RW | 0x0 | dcs_sw_1p_tx This bit configures the DCS short write packet with one parameter command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 16 | RW | 0x0 | dcs_sw_0p_tx This bit configures the DCS short write packet with zero parameter command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 15 | RO | 0x0 | Reserved |
| 14 | RW | 0x0 | gen_lw_tx This bit configures the Generic long write packet command transmission type: 1'b0: High-speed 1'b1: Low-power |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 13 | RW | 0x0 | gen_sr_2p_tx This bit configures the Generic short read packet with two parameters command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 12 | RW | 0x0 | gen_sr_1p_tx This bit configures the Generic short read packet with one parameter command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 11 | RW | 0x0 | gen_sr_0p_tx This bit configures the Generic short read packet with zero parameter command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 10 | RW | 0x0 | gen_sw_2p_tx This bit configures the Generic short write packet with two parameters command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 9 | RW | 0x0 | gen_sw_1p_tx This bit configures the Generic short write packet with one parameter command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 8 | RW | 0x0 | gen_sw_0p_tx This bit configures the Generic short write packet with zero parameter command transmission type: 1'b0: High-speed 1'b1: Low-power |
| 7:2 | RO | 0x0 | Reserved |
| 1 | RW | 0x0 | ack_rqst_en When set to 1, this bit enables the acknowledge request after each packet transmission |
| 0 | RW | 0x0 | tear_fx_en When set to 1, this bit enables the tearing effect acknowledge request |

MIPI DSI HOST GEN HDR

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | Reserved |
| 23:16 | RW | 0x00 | gen_wc_msbyte |
| 15:8 | RW | 0x00 | gen_wc_lsbyte This field configures the least significant byte of the header packet's Word count for long packets or data 0 for short packets |
| 7:6 | RW | 0x0 | gen_vc This field configures the virtual channel id of the header packet |
| 5:0 | RW | 0x00 | gen_dt This field configures the packet data type of the header packet |

MIPI DSI HOST GEN PLD DATA

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RW | 0x00 | gen_pld_b4 This field indicates byte 4 of the packet payload |
| 23:16 | RW | 0x00 | gen_pld_b3 This field indicates byte 3 of the packet payload |
| 15:8 | RW | 0x00 | gen_pld_b2 This field indicates byte 2 of the packet payload |
| 7:0 | RW | 0x00 | gen_pld_b1 This field indicates byte 1 of the packet payload |

MIPI DSI HOST CMD PKT STATUS

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:15 | RO | 0x0 | Reserved |
| 14 | RW | 0x0 | dbi_rd_cmd_busy This bit is set when a read command is issued and cleared when the entire response is stored in the FIFO |
| 13 | RO | 0x0 | dbi_pld_r_full This bit indicates the full status of the DBI read payload FIFO |
| 12 | RO | 0x0 | dbi_pld_r_empty This bit indicates the empty status of the DBI read payload FIFO |
| 11 | RO | 0x0 | dbi_pld_w_full This bit indicates the full status of the DBI write payload FIFO |
| 10 | RO | 0x0 | dbi_pld_w_empty This bit indicates the empty status of the DBI write payload FIFO |
| 9 | RO | 0x0 | dbi_cmd_full This bit indicates the full status of the DBI command FIFO |
| 8 | RO | 0x0 | dbi_cmd_empty This bit indicates the empty status of the DBI command FIFO |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 7 | RO | 0x0 | Reserved |
| 6 | RO | 0x0 | gen_rd_cmd_busy This bit is set when a read command is issued and cleared when the entire response is stored in the FIFO |
| 5 | RO | 0x0 | gen_pld_r_full This bit indicates the full status of the generic read payload FIFO |
| 4 | RO | 0x0 | gen_pld_r_empty This bit indicates the empty status of the generic read payload FIFO |
| 3 | RO | 0x0 | gen_pld_w_full This bit indicates the full status of the generic write payload FIFO |
| 2 | RO | 0x0 | gen_pld_w_empty This bit indicates the empty status of the generic write payload FIFO |
| 1 | RO | 0x0 | gen_cmd_full This bit indicates the full status of the generic command FIFO |
| 0 | RO | 0x0 | gen_cmd_empty This bit indicates the empty status of the generic command FIFO |

MIPI DSI HOST TO CNT CFG

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | <p>hstx_to_cnt This field configures the timeout counter that triggers a high-speed transmission timeout contention detection (measured in TO_CLK_DIVISION cycles). If using the non-burst mode and there is no sufficient time to switch from HS to LP and back in the period which is from one line data finishing to the next line sync start, the DSI link returns the LP state once per frame, then you should configure the TO_CLK_DIVISION and hstx_to_cnt to be in accordance with:</p> <p>$hstx_to_cnt * lanebyteclkperiod * TO_CLK_DIVISION \geq$ the time of one FRAME data transmission * (1 + 10%) In burst mode, RGB pixel packets are time-compressed, leaving more time during a can line. Therefore, if in burst mode and there is sufficient time to switch from HS to LP and back in the period of time from one line data finishing to the next line sync start, the DSI link can return LP mode and back in this time interval to save power. For this, configure the TO_CLK_DIVISION and hstx_to_cnt to be in accordance with:</p> <p>$hstx_to_cnt * lanebyteclkperiod * TO_CLK_DIVISION \geq$ the time of one LINE data transmission * (1 + 10%)</p> |
| 15:0 | RW | 0x0000 | <p>lprx_to_cnt This field configures the timeout counter that triggers a low-power reception timeout contention detection (measured in TO_CLK_DIVISION cycles)</p> |

MIPI DSI HOST HS RD TO CNT

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RW | 0x0000 | <p>hs_rd_to_cnt This field sets a period for which the DWC_mipi_dsi_host keeps the link still, after sending a high-speed read operation. This period is measured in cycles of lanebyteclk. The counting starts when the DPHY enters the Stop state and causes no interrupts</p> |

MIPI DSI HOST LP RD TO CNT

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RW | 0x0000 | lp_rd_to_cnt This field sets a period for which the DWC_mipi_dsi_host keeps the link still, after sending a low-power read operation. This period is measured in cycles of lanebyteclk. The counting starts when the DPHY enters the Stop state and causes no interrupts |

MIPI DSI HOST HS WR TO CNT

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | Reserved |
| 24 | RW | 0x0 | presp_to_mode When set to 1, this bit ensures that the peripheral response timeout caused by hs_wr_to_cnt is used only once per eDPI frame, when both the following conditions are met: dpivsync_edpiwms has risen and fallen. Packets originated from eDPI have been transmitted and its FIFO is empty again. In this scenario no non-eDPI requests are sent to the DPHY, even if there is traffic from generic or DBI ready to be sent, making it return to stop state. When it does so, PRESP_TO counter is activated and only when it finishes does the controller send any other traffic that is ready |
| 23:16 | RO | 0x0 | Reserved |
| 15:0 | RW | 0x0000 | hs_wr_to_cnt This field sets a period for which the DWC_mipi_dsi_host keeps the link inactive after sending a high-speed write operation. This period is measured in cycles of lanebyteclk. The counting starts when the DPHY enters the Stop state and causes no interrupts |

MIPI DSI HOST LP WR TO CNT

Address: Operational Base + offset (0x0088)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RW | 0x0000 | lp_wr_to_cnt This field sets a period for which the DWC_mipi_dsi_host keeps the link still, after sending a low-power write operation. This period is measured in cycles of lanebyteclk. The counting starts when the DPHY enters the Stop state and causes no interrupts |

MIPI DSI HOST BTA TO CNT

Address: Operational Base + offset (0x008c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | bta_to_cnt This field sets a period for which the DWC_mipi_dsi_host keeps the link still, after completing a Bus Turn-Around. This period is measured in cycles of lanebyteclk. The counting starts when the DPHY enters the Stop state and causes no interrupts |

MIPI DSI HOST SDF 3D

Address: Operational Base + offset (0x0090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | Reserved |
| 16 | RW | 0x0 | send_3d_cfg When set, causes the next VSS packet to include 3D control payload in every VSS packet |
| 15:6 | RO | 0x0 | Reserved |
| 5 | RW | 0x0 | right_first This bit defines the left or right order: 1'b0: Left eye data is sent first, and then the right eye data is sent. 1'b1: Right eye data is sent first, and then the left eye data is sent |
| 4 | RW | 0x0 | second_vsync This field defines whether there is a second VSYNC pulse between Left and Right Images, when 3D Image Format is Frame-based: 1'b0: No sync pulses between left and right data 1'b1: Sync pulse (HSYNC, VSYNC, blanking) between left and right data |
| 3:2 | RW | 0x0 | format_3d This field defines the 3D image format: 2'b00: Line (alternating lines of left and right data) 2'b01: Frame (alternating frames of left and right data) 2'b10: Pixel (alternating pixels of left and right data) 2'b11: Reserved |
| 1:0 | RW | 0x0 | mode_3d This field defines the 3D mode on/off and display orientation: 2'b00: 3D mode off (2D mode on) 2'b01: 3D mode on, portrait orientation 2'b10: 3D mode on, landscape orientation 2'b11: Reserved |

MIPI DSI HOST LPCLK CTRL

Address: Operational Base + offset (0x0094)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:2 | RO | 0x0 | Reserved |
| 1 | RW | 0x0 | auto_clklane_ctrl This bit enables the automatic mechanism to stop providing clock in the clock lane when time allows |
| 0 | RW | 0x0 | phy_txrequestclkhs This bit controls the DPHY PPI txrequestclkhs signal |

MIPI DSI HOST PHY TMR LPCLK CFG

Address: Operational Base + offset (0x0098)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31:26 | RO | 0x0 | Reserved |
| 25:16 | RW | 0x000 | phy_clkhs2lp_time This field configures the maximum time that the DPHY clock lane takes to go from high-speed to low-power transmission measured in lane byte clock cycles |
| 15:10 | RO | 0x0 | Reserved |
| 9:0 | RW | 0x000 | phy_clklp2hs_time This field configures the maximum time that the DPHY clock lane takes to go from low-power to high-speed transmission measured in lane byte clock cycles |

MIPI DSI HOST PHY TMR CFG

Address: Operational Base + offset (0x009c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:24 | RW | 0x00 | phy_hs2lp_time This field configures the maximum time that the DPHY data lanes take to go from high-speed to low-power transmission measured in lane byte clock cycles |
| 23:16 | RW | 0x00 | phy_lp2hs_time This field configures the maximum time that the DPHY data lanes take to go from low-power to high-speed transmission measured in lane byte clock cycles |
| 15 | RO | 0x0 | Reserved |
| 14:0 | RW | 0x0000 | max_rd_time This field configures the maximum time required to perform a read command in lane byte clock cycles. This register can only be modified when no read command is in progress |

MIPI DSI HOST PHY RSTZ

Address: Operational Base + offset (0x00a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RO | 0x0 | Reserved |
| 3 | RW | 0x0 | phy_forcepll When the DPHY is in ULPS, this bit enables the DPHY PLL. Dependency: DSI_HOST_FPGA = 0. Otherwise, this bit is Reserved |
| 2 | RW | 0x0 | phy_enableclk When set to 1, this bit enables the DPHY Clock Lane module |
| 1 | RW | 0x0 | phy_rstz When set to 0, this bit places the digital section of the DPHY in the reset state |
| 0 | RW | 0x0 | phy_shutdownz When set to 0, this bit places the DPHY macro in power-down state |

MIPI DSI HOST PHY IF CFG

Address: Operational Base + offset (0x00a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | phy_stop_wait_time This field configures the minimum wait period to request a high-speed transmission after the Stop state |
| 7:2 | RO | 0x0 | Reserved |
| 1:0 | RW | 0x3 | n_lanes This field configures the number of active data lanes: 2'b00: One data lane (lane 0) 2'b01: Two data lanes (lanes 0 and 1) 2'b10: Three data lanes (lanes 0, 1, and 2) 2'b11: Four data lanes (lanes 0, 1, 2, and 3) |

MIPI DSI HOST PHY STATUS

Address: Operational Base + offset (0x00b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | Reserved |
| 12 | RO | 0x0 | phy_ulpsactivenot3lane This bit indicates the status of ulpsactivenot3lane DPHY signal |
| 11 | RO | 0x0 | phy_stopstate3lane This bit indicates the status of phystopstate3lane DPHY signal |
| 10 | RO | 0x0 | phy_ulpsactivenot2lane This bit indicates the status of ulpsactivenot2lane DPHY signal |
| 9 | RO | 0x0 | phy_stopstate2lane This bit indicates the status of phystopstate2lane DPHY signal |
| 8 | RO | 0x0 | phy_ulpsactivenot1lane This bit indicates the status of ulpsactivenot1lane DPHY signal |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7 | RO | 0x0 | phy_stopstate1lane This bit indicates the status of phystopstate1lane DPHY signal |
| 6 | RO | 0x0 | phy_rxulpsesc0lane This bit indicates the status of rxulpsesc0lane DPHY signal |
| 5 | RO | 0x0 | phy_ulpssactivenot0lane This bit indicates the status of ulpsactivenot0lane DPHY signal |
| 4 | RO | 0x0 | phy_stopstate0lane This bit indicates the status of phystopstate0lane DPHY signal |
| 3 | RO | 0x0 | phy_ulpssactivenotclk This bit indicates the status of phyulpssactivenotclk DPHY signal |
| 2 | RO | 0x0 | phy_stopstateclklane This bit indicates the status of phystopstateclklane DPHY signal |
| 1 | RO | 0x0 | phy_direction This bit indicates the status of phydirection DPHY signal |
| 0 | RO | 0x0 | phy_lock This bit indicates the status of phylock DPHY signal |

MIPI DSI HOST INT ST0

Address: Operational Base + offset (0x00bc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:21 | RO | 0x0 | Reserved |
| 20 | RO | 0x0 | dphy_errors_4 This bit indicates the LP1 contention error ErrContentionLP1 from Lane 0 |
| 19 | RW | 0x0 | dphy_errors_3 This bit indicates the LP0 contention error ErrContentionLP0 from Lane 0 |
| 18 | RO | 0x0 | dphy_errors_2 This bit indicates the ErrControl error from Lane 0 |
| 17 | RO | 0x0 | dphy_errors_1 This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0 |
| 16 | RO | 0x0 | dphy_errors_0 This bit indicates ErrEsc escape entry error from Lane 0 |
| 15 | RO | 0x0 | ack_with_err_15 This bit retrieves the DSI protocol violation from the Acknowledge error report |
| 14 | RO | 0x0 | ack_with_err_14 This bit retrieves the Reserved (specific to device) from the Acknowledge error report |
| 13 | RO | 0x0 | ack_with_err_13 This bit retrieves the invalid transmission length from the Acknowledge error report |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12 | RO | 0x0 | ack_with_err_12 This bit retrieves the DSI VC ID Invalid from the Acknowledge error report |
| 11 | RO | 0x0 | ack_with_err_11 This bit retrieves the not recognized DSI data type from the Acknowledge error report |
| 10 | RO | 0x0 | ack_with_err_10 This bit retrieves the checksum error (long packet only) from the Acknowledge error report |
| 9 | RO | 0x0 | ack_with_err_9 This bit retrieves the ECC error, multi-bit (detected, not corrected) from the Acknowledge error |
| 8 | RO | 0x0 | ack_with_err_8 This bit retrieves the ECC error, single-bit (detected and corrected) from the Acknowledge error |
| 7 | RO | 0x0 | ack_with_err_7 This bit retrieves the Reserved (specific to device) from the Acknowledge error report |
| 6 | RO | 0x0 | ack_with_err_6 This bit retrieves the False Control error from the Acknowledge error report |
| 5 | RO | 0x0 | ack_with_err_5 This bit retrieves the Peripheral Timeout error from the Acknowledge Error report |
| 4 | RO | 0x0 | ack_with_err_4 This bit retrieves the LP Transmit Sync error from the Acknowledge error report |
| 3 | RO | 0x0 | ack_with_err_3 This bit retrieves the Escape Mode Entry Command error from the Acknowledge error report |
| 2 | RO | 0x0 | ack_with_err_2 This bit retrieves the EoT Sync error from the Acknowledge error report |
| 1 | RO | 0x0 | ack_with_err_1 This bit retrieves the SoT Sync error from the Acknowledge error report |
| 0 | RO | 0x0 | ack_with_err_0 This bit retrieves the SoT error from the Acknowledge error report |

MIPI DSI HOST INT ST1

Address: Operational Base + offset (0x00c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:18 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17 | RW | 0x0 | dbi_illegal_comm_err This bit indicates that an attempt to write an illegal command on the DBI interface is made and the core is blocked by transmission |
| 16 | RO | 0x0 | dbi_pld_recv_err This bit indicates that during a DBI read back packet, the payload FIFO becomes full and the received data is corrupted |
| 15 | RO | 0x0 | dbi_pld_rd_err This bit indicates that during a DCS read data, the payload FIFO goes empty and the data sent to the interface is corrupted |
| 14 | RO | 0x0 | dbi_pld_wr_err This bit indicates that the system tried to write the payload data through the DBI interface and the FIFO is full. Therefore, the command is not written |
| 13 | RO | 0x0 | dbi_cmd_wr_err This bit indicates that the system tried to write a command through the DBI but the command FIFO is full. Therefore, the command is not written |
| 12 | RO | 0x0 | gen_pld_recev_err This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted |
| 11 | RO | 0x0 | gen_pld_rd_err This bit indicates that during a DCS read data, the payload FIFO becomes empty and the data sent to the interface is corrupted |
| 10 | RO | 0x0 | gen_pld_send_err This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent |
| 9 | RO | 0x0 | gen_pld_wr_err This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the payload is not written |
| 8 | RO | 0x0 | gen_cmd_wr_err This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written |
| 7 | RO | 0x0 | dpi_pld_wr_err This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted |
| 6 | RO | 0x0 | eotp_err This bit indicates that the EoTp packet is not received at the end of the incoming peripheral transmission |
| 5 | RO | 0x0 | pkt_size_err This bit indicates that the packet size error is detected during the packet reception |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 4 | RO | 0x0 | crc_err This bit indicates that the CRC error is detected in the received packet payload |
| 3 | RO | 0x0 | ecc_multi_err This bit indicates that the ECC multiple error is detected in a received packet |
| 2 | RO | 0x0 | ecc_single_err This bit indicates that the ECC single error is detected and corrected in a received packet |
| 1 | RO | 0x0 | to_lp_rx This bit indicates that the low-power reception timeout counter reached the end and contention is detected |
| 0 | RO | 0x0 | to_hs_tx This bit indicates that the high-speed transmission timeout counter reached the end and contention is detected |

MIPI DSI HOST INT MSK0

Address: Operational Base + offset (0x00c4)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|--|
| 31:21 | RO | 0x0 | Reserved |
| 20 | RW | 0x0 | dphy_errors_4 This bit indicates the LP1 contention error ErrContentionLP1 from Lane 0 |
| 19 | RW | 0x0 | dphy_errors_3 This bit indicates the LP0 contention error ErrContentionLP0 from Lane 0 |
| 18 | RW | 0x0 | dphy_errors_2 This bit indicates the ErrControl error from Lane 0 |
| 17 | RW | 0x0 | dphy_errors_1 This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0 |
| 16 | RW | 0x0 | dphy_errors_0 This bit indicates ErrEsc escape entry error from Lane 0 |
| 15 | RW | 0x0 | ack_with_err_15 This bit retrieves the DSI protocol violation from the Acknowledge error report |
| 14 | RW | 0x0 | ack_with_err_14 This bit retrieves the Reserved (specific to device) from the Acknowledge error report |
| 13 | RW | 0x0 | ack_with_err_13 This bit retrieves the invalid transmission length from the Acknowledge error report |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12 | RW | 0x0 | ack_with_err_12 This bit retrieves the DSI VC ID Invalid from the Acknowledge error report |
| 11 | RW | 0x0 | ack_with_err_11 This bit retrieves the not recognized DSI data type from the Acknowledge error report |
| 10 | RW | 0x0 | ack_with_err_10 This bit retrieves the checksum error (long packet only) from the Acknowledge error report |
| 9 | RW | 0x0 | ack_with_err_9 This bit retrieves the ECC error, multi-bit (detected, not corrected) from the Acknowledge error |
| 8 | RW | 0x0 | ack_with_err_8 This bit retrieves the ECC error, single-bit (detected and corrected) from the Acknowledge error |
| 7 | RW | 0x0 | ack_with_err_7 This bit retrieves the Reserved (specific to device) from the Acknowledge error report |
| 6 | RW | 0x0 | ack_with_err_6 This bit retrieves the False Control error from the Acknowledge error report |
| 5 | RW | 0x0 | ack_with_err_5 This bit retrieves the Peripheral Timeout error from the Acknowledge Error report |
| 4 | RW | 0x0 | ack_with_err_4 This bit retrieves the LP Transmit Sync error from the Acknowledge error report |
| 3 | RW | 0x0 | ack_with_err_3 This bit retrieves the Escape Mode Entry Command error from the Acknowledge error report |
| 2 | RW | 0x0 | ack_with_err_2 This bit retrieves the EoT Sync error from the Acknowledge error report |
| 1 | RW | 0x0 | ack_with_err_1 This bit retrieves the SoT Sync error from the Acknowledge error report |
| 0 | RW | 0x0 | ack_with_err_0 This bit retrieves the SoT error from the Acknowledge error report |

MIPI DSI HOST INT MSK1

Address: Operational Base + offset (0x00c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:18 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17 | RW | 0x0 | dbi_illegal_comm_err This bit indicates that an attempt to write an illegal command on the DBI interface is made and the core is blocked by transmission |
| 16 | RW | 0x0 | dbi_pld_recv_err This bit indicates that during a DBI read back packet, the payload FIFO becomes full and the received data is corrupted |
| 15 | RW | 0x0 | dbi_pld_rd_err This bit indicates that during a DCS read data, the payload FIFO goes empty and the data sent to the interface is corrupted |
| 14 | RW | 0x0 | dbi_pld_wr_err This bit indicates that the system tried to write the payload data through the DBI interface and the FIFO is full. Therefore, the command is not written |
| 13 | RW | 0x0 | dbi_cmd_wr_err This bit indicates that the system tried to write a command through the DBI but the command FIFO is full. Therefore, the command is not written |
| 12 | RW | 0x0 | gen_pld_recev_err This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted |
| 11 | RW | 0x0 | gen_pld_rd_err This bit indicates that during a DCS read data, the payload FIFO becomes empty and the data sent to the interface is corrupted |
| 10 | RW | 0x0 | gen_pld_send_err This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent |
| 9 | RW | 0x0 | gen_pld_wr_err This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the payload is not written |
| 8 | RW | 0x0 | gen_cmd_wr_err This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written |
| 7 | RW | 0x0 | dpi_pld_wr_err This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted |
| 6 | RW | 0x0 | eotp_err This bit indicates that the EoTp packet is not received at the end of the incoming peripheral transmission |
| 5 | RW | 0x0 | pkt_size_err This bit indicates that the packet size error is detected during the packet reception |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 4 | RW | 0x0 | crc_err This bit indicates that the CRC error is detected in the received packet payload |
| 3 | RW | 0x0 | ecc_multi_err This bit indicates that the ECC multiple error is detected in a received packet |
| 2 | RW | 0x0 | ecc_single_err This bit indicates that the ECC single error is detected and corrected in a received packet |
| 1 | RW | 0x0 | to_lp_rx This bit indicates that the low-power reception timeout counter reached the end and contention is detected |
| 0 | RW | 0x0 | to_hs_tx This bit indicates that the high-speed transmission timeout counter reached the end and contention is detected |

MIPI DSI HOST INT FORCE0

Address: Operational Base + offset (0x00d8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:21 | RO | 0x0 | Reserved |
| 20 | RW | 0x0 | dphy_errors_4 This bit indicates the LP1 contention error ErrContentionLP1 from Lane 0 |
| 19 | RW | 0x0 | dphy_errors_3 This bit indicates the LP0 contention error ErrContentionLP0 from Lane 0 |
| 18 | RW | 0x0 | dphy_errors_2 This bit indicates the ErrControl error from Lane 0 |
| 17 | RW | 0x0 | dphy_errors_1 This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0 |
| 16 | RW | 0x0 | dphy_errors_0 This bit indicates ErrEsc escape entry error from Lane 0 |
| 15 | RW | 0x0 | ack_with_err_15 This bit retrieves the DSI protocol violation from the Acknowledge error report |
| 14 | RW | 0x0 | ack_with_err_14 This bit retrieves the Reserved (specific to device) from the Acknowledge error report |
| 13 | RW | 0x0 | ack_with_err_13 This bit retrieves the invalid transmission length from the Acknowledge error report |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12 | RW | 0x0 | ack_with_err_12 This bit retrieves the DSI VC ID Invalid from the Acknowledge error report |
| 11 | RW | 0x0 | ack_with_err_11 This bit retrieves the not recognized DSI data type from the Acknowledge error report |
| 10 | RW | 0x0 | ack_with_err_10 This bit retrieves the checksum error (long packet only) from the Acknowledge error report |
| 9 | RW | 0x0 | ack_with_err_9 This bit retrieves the ECC error, multi-bit (detected, not corrected) from the Acknowledge error |
| 8 | RW | 0x0 | ack_with_err_8 This bit retrieves the ECC error, single-bit (detected and corrected) from the Acknowledge error |
| 7 | RW | 0x0 | ack_with_err_7 This bit retrieves the Reserved (specific to device) from the Acknowledge error report |
| 6 | RW | 0x0 | ack_with_err_6 This bit retrieves the False Control error from the Acknowledge error report |
| 5 | RW | 0x0 | ack_with_err_5 This bit retrieves the Peripheral Timeout error from the Acknowledge Error report |
| 4 | RW | 0x0 | ack_with_err_4 This bit retrieves the LP Transmit Sync error from the Acknowledge error report |
| 3 | RW | 0x0 | ack_with_err_3 This bit retrieves the Escape Mode Entry Command error from the Acknowledge error report |
| 2 | RW | 0x0 | ack_with_err_2 This bit retrieves the EoT Sync error from the Acknowledge error report |
| 1 | RW | 0x0 | ack_with_err_1 This bit retrieves the SoT Sync error from the Acknowledge error report |
| 0 | RW | 0x0 | ack_with_err_0 This bit retrieves the SoT error from the Acknowledge error report |

MIPI DSI HOST INT FORCE1

Address: Operational Base + offset (0x00dc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:18 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17 | RW | 0x0 | dbi_illegal_comm_err This bit indicates that an attempt to write an illegal command on the DBI interface is made and the core is blocked by transmission |
| 16 | RW | 0x0 | dbi_pld_recv_err This bit indicates that during a DBI read back packet, the payload FIFO becomes full and the received data is corrupted |
| 15 | RW | 0x0 | dbi_pld_rd_err This bit indicates that during a DCS read data, the payload FIFO goes empty and the data sent to the interface is corrupted |
| 14 | RW | 0x0 | dbi_pld_wr_err This bit indicates that the system tried to write the payload data through the DBI interface and the FIFO is full. Therefore, the command is not written |
| 13 | RW | 0x0 | dbi_cmd_wr_err This bit indicates that the system tried to write a command through the DBI but the command FIFO is full. Therefore, the command is not written |
| 12 | RW | 0x0 | gen_pld_recev_err This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted |
| 11 | RW | 0x0 | gen_pld_rd_err This bit indicates that during a DCS read data, the payload FIFO becomes empty and the data sent to the interface is corrupted |
| 10 | RW | 0x0 | gen_pld_send_err This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent |
| 9 | RW | 0x0 | gen_pld_wr_err This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the payload is not written |
| 8 | RW | 0x0 | gen_cmd_wr_err This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written |
| 7 | RW | 0x0 | dpi_pld_wr_err This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted |
| 6 | RW | 0x0 | eotp_err This bit indicates that the EoTp packet is not received at the end of the incoming peripheral transmission |
| 5 | RW | 0x0 | pkt_size_err This bit indicates that the packet size error is detected during the packet reception |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 4 | RW | 0x0 | crc_err This bit indicates that the CRC error is detected in the received packet payload |
| 3 | RW | 0x0 | ecc_multi_err This bit indicates that the ECC multiple error is detected in a received packet |
| 2 | RW | 0x0 | ecc_single_err This bit indicates that the ECC single error is detected and corrected in a received packet |
| 1 | RW | 0x0 | to_lp_rx This bit indicates that the low-power reception timeout counter reached the end and contention is detected |
| 0 | RW | 0x0 | to_hs_tx This bit indicates that the high-speed transmission timeout counter reached the end and contention is detected |

MIPI DSI HOST VID SHADOW CTRL

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:17 | RO | 0x0 | Reserved |
| 16 | RW | 0x0 | vid_shadow_pin_req When set to 1, the video request is done by external pin. In this mode, vid_shadow_req is ignored |
| 15:9 | RO | 0x0 | Reserved |
| 8 | RW | 0x0 | vid_shadow_req When set to 1, the DPI registers are copied to the auxiliary registers. After copying, this bit is auto cleared |
| 7:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | vid_shadow_en When set to 1, DPI receives the active configuration from the auxiliary registers. When this bit is set along with the id_shadow_req bit, the auxiliary registers are automatically updated |

MIPI DSI HOST DPI VCID ACT

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:2 | RO | 0x0 | Reserved |
| 1:0 | RW | 0x0 | dpi_vcidx This field configures the DPI virtual channel id that is indexed to the Video mode packets |

MIPI DSI HOST DPI COLOR CODING ACT

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:9 | RO | 0x0 | Reserved |
| 8 | RW | 0x0 | loosely18_en When set to 1, this bit activates loosely packed variant to 18-bit configurations |
| 7:4 | RO | 0x0 | Reserved |
| 3:0 | RW | 0x0 | dpi_color_coding This field configures the DPI color coding as follows: 4'b0000: 16-bit configuration 1 4'b0001: 16-bit configuration 2 4'b0010: 16-bit configuration 3 4'b0011: 18-bit configuration 1 4'b0100: 18-bit configuration 2 4'b0101: 24-bit 4'b0110: 20-bit YCbCr 4:2:2 loosely packed 4'b0111: 24-bit YCbCr 4:2:2 4'b1000: 16-bit YCbCr 4:2:2 4'b1001: 30-bit 4'b1010: 36-bit 4'b1011-4'b1111: 12-bit YCbCr 4:2:0 |

MIPI DSI HOST DPI LP CMD TIM ACT

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:24 | RO | 0x0 | Reserved |
| 23:16 | RO | 0x00 | outvact_lpcmd_time This field is used for the transmission of commands in low-power mode. It defines the size, in bytes, of the largest packet that can fit in a line during the VSA, VBP, and VFP regions |
| 15:8 | RO | 0x0 | Reserved |
| 7:0 | RO | 0x00 | invact_lpcmd_time This field is used for the transmission of commands in low-power mode. It defines the size, in bytes, of the largest packet that can fit in a line during the VACT region |

MIPI DSI HOST VID MODE CFG ACT

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | Reserved |
| 24 | RW | 0x0 | vpg_orientation This field indicates the color bar orientation as follows: 1'b0: Vertical mode 1'b1: Horizontal mode |
| 23:21 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 20 | RO | 0x0 | vpg_mode This field is to select the pattern: 1'b0: Color bar (horizontal or vertical) 1'b1: BER pattern (vertical only) |
| 19:17 | RO | 0x0 | Reserved |
| 16 | RO | 0x0 | vpg_en When set to 1, this bit enables the video mode pattern generator |
| 15 | RO | 0x0 | lp_cmd_en When set to 1, this bit enables the command transmission only in lowpower mode |
| 14 | RO | 0x0 | frame_bta_ack_en When set to 1, this bit enables the request for an acknowledge response at the end of a frame |
| 13 | RO | 0x0 | lp_hfp_en When set to 1, this bit enables the return to low-power inside the Horizontal Front Porch (HFP) period when timing allows |
| 12 | RO | 0x0 | lp_hbp_en When set to 1, this bit enables the return to low-power inside the Horizontal Back Porch (HBP) period when timing allows |
| 11 | RO | 0x0 | lp_vact_en When set to 1, this bit enables the return to low-power inside the Vertical Active (VACT) period when timing allows |
| 10 | RO | 0x0 | lp_vfp_en When set to 1, this bit enables the return to low-power inside the Vertical Front Porch (VFP) period when timing allows |
| 9 | RO | 0x0 | lp_vbp_en When set to 1, this bit enables the return to low-power inside the Vertical Back Porch (VBP) period when timing allows |
| 8 | RO | 0x0 | lp_vsa_en When set to 1, this bit enables the return to low-power inside the Vertical Sync Time (VSA) period when timing allows |
| 7:2 | RO | 0x0 | Reserved |
| 1:0 | RO | 0x0 | vid_mode_type This field indicates the video mode transmission type as follows: 2'b00: Non-burst with sync pulses 2'b01: Non-burst with sync events 2'b10 and 2'b11: Burst mode |

MIPI DSI HOST VID PKT SIZE ACT

Address: Operational Base + offset (0x013c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:14 | RO | 0x0 | Reserved |
| 13:0 | RO | 0x0000 | vid_pkt_size This field configures the number of pixels in a single video packet. For 18-bit not loosely packed data types, this number must be a multiple of 4. For YCbCr data types, it must be a multiple of 2, as described in the DSI specification |

MIPI DSI HOST VID NUM CHUNKS ACT

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:13 | RO | 0x0 | Reserved |
| 12:0 | RO | 0x0000 | vid_num_chunks This register configures the number of chunks to be transmitted during a Line period (a chunk consists of a video packet and a null packet). If set to 0 or 1, the video line is transmitted in a single packet. If set to 1, the packet is part of a chunk, so a null packet follows it if vid_null_size > 0. Otherwise, multiple chunks are used to transmit each video line |

MIPI DSI HOST VID NULL SIZE ACT

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x0000 | vid_null_size This register configures the number of bytes inside a null packet. Setting it to 0 disables the null packets |

MIPI DSI HOST VID HSA TIME ACT

Address: Operational Base + offset (0x0148)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | Reserved |
| 11:0 | RO | 0x000 | vid_hsa_time This field configures the Horizontal Synchronism Active period in lane byte clock cycles |

MIPI DSI HOST VID HBP TIME ACT

Address: Operational Base + offset (0x014c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:12 | RO | 0x0 | Reserved |
| 11:0 | RW | 0x000 | vid_hsa_time This field configures the Horizontal Synchronism Active period in lane byte clock cycles |

MIPI DSI HOST VID HLINE TIME ACT

Address: Operational Base + offset (0x0150)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:15 | RO | 0x0 | Reserved |
| 14:0 | RO | 0x0000 | vid_hline_time This field configures the size of the total line time (HSA+HBP+HACT+HFP) counted in lane byte clock cycles |

MIPI DSI HOST VID VSA LINES ACT

Address: Operational Base + offset (0x0154)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | Reserved |
| 9:0 | RO | 0x000 | vsa_lines This field configures the Vertical Synchronism Active period measured in number of horizontal lines |

MIPI DSI HOST VID VBP LINES ACT

Address: Operational Base + offset (0x0158)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:10 | RO | 0x0 | Reserved |
| 9:0 | RW | 0x000 | vbp_lines This field configures the Vertical Back Porch period measured in number of horizontal lines |

MIPI DSI HOST VID VFP LINES ACT

Address: Operational Base + offset (0x015c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RO | 0x0 | vfp_lines This field configures the Vertical Front Porch period measured in number of horizontal lines |

MIPI DSI HOST VID VACTIVE LINES ACT

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:14 | RO | 0x0 | Reserved |
| 13:0 | RO | 0x0000 | v_active_lines This field configures the Vertical Active period measured in number of horizontal lines |

MIPI DSI HOST SDF 3D ACT

Address: Operational Base + offset (0x0190)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | Reserved |
| 16 | RW | 0x0 | send_3d_cfg When set, causes the next VSS packet to include 3D control payload in every VSS packet |
| 15:6 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|---|
| 5 | RO | 0x0 | right_first This bit defines the left or right order: 1'b0: Left eye data is sent first, and then the right eye data is sent. 1'b1: Right eye data is sent first, and then the left eye data is sent |
| 4 | RO | 0x0 | second_vsync This field defines whether there is a second VSYNC pulse between Left and Right Images, when 3D Image Format is Frame-based: 1'b0: No sync pulses between left and right data 1'b1: Sync pulse (HSYNC, VSYNC, blanking) between left and right data |
| 3:2 | RO | 0x0 | format_3d This field defines the 3D image format: 2'b00: Line (alternating lines of left and right data) 2'b01: Frame (alternating frames of left and right data) 2'b10: Pixel (alternating pixels of left and right data) 2'b11: Reserved |
| 1:0 | RO | 0x0 | mode_3d This field defines the 3D mode on/off and display orientation: 2'b00: 3D mode off (2D mode on) 2'b01: 3D mode on, portrait orientation 2'b10: 3D mode on, landscape orientation 2'b11: Reserved |

12.5 Application Notes

12.5.1 COMMON CONFIGURATION (DEFAULT IN MIPI MODE)

Step1: Enable reference clock.

Step2: Wait a period after reference clock have been enabled.

Step3: Send 0xe4 to register MIPI_DSI_DPHY_CTRL_PWRCTL. Enable PLL and LDO.

Step4: Send 0x7d to register MIPI_DSI_DPHY_CTRL_LANE_ENABLE. Enable all lanes on analog part.

Step5: Send 0xe0 to register MIPI_DSI_DPHY_CTRL_PWRCTL. Reset analog.

Step6: Wait a period after analog has been reset.

Step7: Send 0x1e to register MIPI_DSI_DPHY_CTRL_DIG_RST. Reset digital.

Step8: Send 0x1f to register MIPI_DSI_DPHY_CTRL_DIG_RST. Reset digital.

Step9: Wait a period until pll locked. Run in MIPI mode.

12.5.2 LOW POWER MODE (FOR DSI ONLY)

Low Power Mode is a special feature for DPHY. You can control this function by using proper registers from the DPHY with few operations. The following is a step by step instruction for low power mode in and out.

Perform the following steps to configure the DPI packet transmission:

Step1: Global configuration:

Configure n_lanes (PHY_IF_CFG-[1:0]) to define the number of lanes in which the controller has to perform high-speed transmissions.

Step2: Configure the DPI Interface to define how the DPI interface interacts with the controller.

Configure dpi_vid (DPI_CFG-[1:0]): This field configures the virtual channel that the packet generated by the DPI interface is indexed to.

Configure dpi_color_coding (DPI_CFG-[4:2]): This field configures the bits per pixels that the interface transmits and also the variant configuration of each bpp. If you select 18 bpp, and the Enable_18_loosely_packed is not active, the number of pixels per line should be a multiple of four.

Configure dataen_active_low (DPI_CFG-[5]): This bit configures the polarity of the dpidataen signal and enables if it is active low.

Configure vsync_active_low(DPI_CFG-[6]): This bit configures the polarity of the dpivsync signal and enables if it is active low.

Configure vsync_active_low(DPI_CFG-[7]): This bit configures the polarity of the dpivsync signal and enables if it is active low.

Configure vsync_active_low(DPI_CFG-[8]): This bit configures the polarity of the dpishutdn signal and enables if it is active low.

Configure vsync_active_low(DPI_CFG-[9]): This bit configures the polarity of the dpicolorlm signal and enables if it is active low.

Configure en18_loosely(DPI_CFG-[10]): This bit configures if the pixel packing is done loosely or packed when dpi_color_coding is 18 bpp. This bit enables loosely packing.

Step3: Select the Video Transmission Mode to define how the processor requires the video line to be transported through the DSI link.

Configure low-power transitions (VID_MODE_CFG-[8:3]): This defines the video line to be transported through the DSI link.

Configure low-power transitions (VID_MODE_CFG-[8:3]): This defines the video periods which are permitted to go to low-power if there is available time to do so.

Configure frame_BTA_ack (VID_MODE_CFG-[11]): This specifies if the controller should request the peripheral acknowledge message at the end of frames.

Burst mode: In this mode, the entire active pixel line is buffered into a FIFO and transmitted in a single packed with no interruptions. This transmission mode requires that the DPI Pixel FIFO has the capacity to store a full line of active pixel data inside it. This mode is optimally used if the difference between pixel required bandwidth and DSI link bandwidth is very different. This enables the DWC_mipi_dsi_host to quickly dispatch the entire active video line in a single burst of data and then return to low-power mode.

Configure the register fiedl vid_mode_type (VID_MODE_CFG-[10]), num_chunks (VID_PKT_CFG-[20:11]), and null_pkt_size (VID_PKT_CFG-[30:21]) are automatically ignored by the DWC_mipi_dsi_host.

Non-Burst mode: In this mode, the processor uses the partitioning properties of the DWC_mipi_dsi_host to divide the video line transmission into several DSI packets. This is done to match the pixel required bandwidth with the DSI link bandwidth. With this mode, the controller configuration does not require a full line of pixel data to be stored inside the DPI Pixel FIFO. It requires only the content of one video packet.

Configure the vid_mode_type field (VID_MODE_CFG-[2:1]) with 2'b0x.

Configure the vid_mode_type field (VID_MODE_CFG-[2:1]) with 2'b00x to enable the transmission of sync pulses.

Configure the vid_mode_type field (VID_MODE_CFG-[2:1]) with 2'b01 to enable the transmission of sync events.

Configure the vid_mode_type field (VID_MODE_CFG-[10:0]) with the number of pixels to be transmitted in a single packet.

Configure the en_multi_pkt field (VID_MODE_CFG-[9]) to enable the division of the active video transmission into more than one packet.

Configure the num_chunks field (VID_MODE_CFG-[20:11]) with the number of video chunks that the active video transmission is divided into.

Configure the en_null_pkt field (VID_MODE_CFG-[10]) to enable the insertion of null packets between video packets.

The field is effective only when en_multi_pkt field is activated, otherwise the controller ignores it and does not send the null packets.

Configure the null_pkt_size field (VID_MODE_CFG-[30:21]) with the actual size of the inserted null packet.

Step4: Define the DPI Horizontal timing configuration as follows:

Configure the hline_time field (TMR_LINE_CFG-[31:18]) with the time taken by a DPI video

line accounted in Clock Lane bytes clock cycles (for a clock lane at 500 MHz the Lane byte clock period is 8 ns). When the DPI clock and Clock Lane clock are not multiples, the hline_time is a result of a round of a number. If the DWC_mipi_dsi_host is configured to go to low-power, it is possible that the error included in a line is incremented with the next one. At the end of several lines, the DWC_mipi_dsi_host can have a number of errors that can cause a malfunction of the video transmission.

Configure the hsa_time field (TMR_LINE_CFG-[8:0]) with the time taken by a DPI Horizontal Sync Active period accounted in Clock Lane byte clock cycles (normally a period of 8ns).

Configure the hbp_time field (TMR_LINE_CFG-[17:9]) with the time taken by a DPI Horizontal Sync Active period accounted in Clock Lane byte clock cycles (normally a period of 8ns). Special attention should be given to the calculation of this parameter.

Step5: Define the Vertical line configuration:

Configure the vsa_lines field (VTIMING_CFG-[3:0]) with the number of lines existing in the DPI Vertical Sync Active period.

Configure the vbp_lines field (VTIMING_CFG-[9:4]) with the number of lines existing in the DPI Vertical Back Porch period.

Configure the vfp_lines field (VTIMING_CFG-[15:10]) with the number of lines existing in the DPI Vertical Front Porch period.

Configure the v_active_lines field (VTIMING_CFG-[26:16]) with the number of lines existing in the DPI Vertical Active period.

Chapter 13 MIPI CSI Transmitter Device(CSI2TX)

13.1 Overview

MIPI CSI Transmitter is used in mobile and high-speed serial applications where a camera can send the video data using it over MIPI lines to the MIPI CSI Receiver for decoding the data and use it for subsequent processing . MIPI CSI Transmitter adheres to MIPI CSI Specification .The MIPI CSI Transmitter along with DPHY provides a complete solution for encoding MIPI data .

The features listed below which may or may not be present in actual product, may be subject to the third party licensing requirements. Please contact Rockchip for actual product feature configurations and licensing requirements.

CSI2TX supports the following features:

- Support APB2.0 Slave interface
- Support VOP interface
 - Format : RAW8,RAW10
 - Data type , word count , virtual channel configuration
- Support IDI interface
 - Format : RAW8,RAW10, RAW12
 - Data type , word count , virtual channel configuration
 - Support D-GAIN
 - Support DPCC
- Compliant with MIPI CSI-2 Spec v1.3 specification
 - Compliant with MIPI DPHY V1.2
 - Lane configuration: programmable 1 , 2 or 4 Data Lane Configuration
 - Operate in continuous/non-continuous clock modes
 - Data Rate: up to 2.0Gbps per lane

13.2 Block Diagram

CSI2TX comprises with:

Gen_IDI:

The interface receives 32-bit data from IDI interface and groups the data in 128-bit

Interface_mux:

To switch idi and vop interface

Cam_interface:

The interface from the camera is though sync , data , and data enable signals . This module fetches the data from camera interface , groups the data in 128-bit format and places it in the payload FIFO

Header_AFIFO:

This is an asynchronous FIFO that stores the header of packet

Payload_AFIFO:

This is an asynchronous FIFO , which stores the pixel to byte converted packed sensor data in to 128-bit for internal processing . This buffer also helps in handing delay and difference in clock rates of write and read domains .

Packetizer:

On the transmitter side of a CSI-2 Link , Packetizer converts parallel data and signal events are converted into packets . The packetizer appends packet-protocol information and headers , and then sends complete bytes through the Lane Management layer to the DPHY .

Tx_FIFO:

Sensor data is stored in this synchronous configurable FIFO before getting distributed in to the lanes based on configuration . This FIFO handle the latency which arises in DHY to start the byte transfer .

LANE_Distributor:

Sensor data is sent across single/multiple lanes based on the lane configuration register

REG_File:

This block holds the complete registers required for CSI transmitter to be operational . A APB interface is provided to control the registers . This module works in the apb_clk domain .

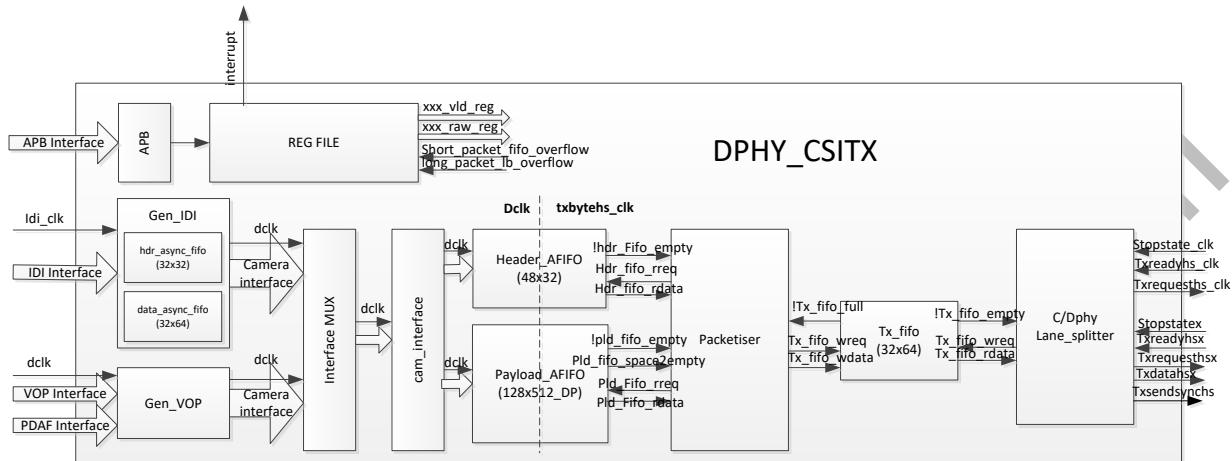


Fig. 13-1 CSI2TX Block Diagram

13.3 Function Description

D-gain:

$$dgain_{out} = (dgain_{in} - sw_black_level) * sw_d_gain + sw_black_level$$

DPCC:

- Support raw8/raw10/raw12
- 4 dpcc mode
- The max size of dpcc sub-block is 256x256
- Up to 16 points in one sub-block

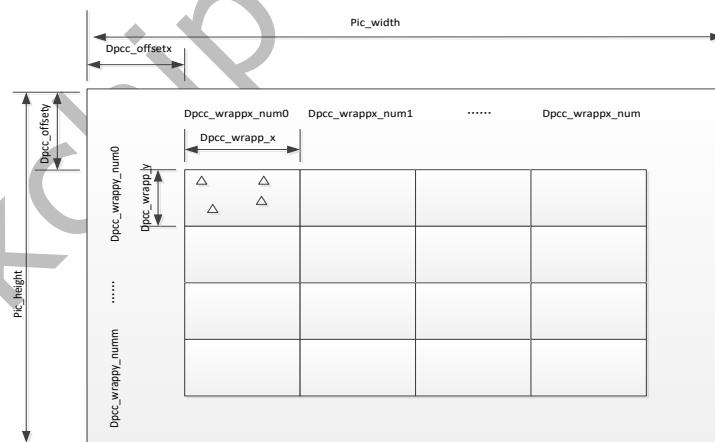


Fig. 13-2 DPCC pattern

13.4 Register Description

13.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

13.4.2 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|-------------------------------|--------|------|-------------|----------------------------|
| CSITX CONFIG DONE | 0x0000 | W | 0x00000000 | CSITX config done |
| CSITX CSITX EN | 0x0004 | W | 0x00000000 | CSITX enable |
| CSITX CSITX VERSION | 0x0008 | W | 0x00000000 | CSITX version |
| CSITX SYS CTRL0 IMD | 0x0010 | W | 0x00000000 | System control0 |
| CSITX SYS CTRL1 | 0x0014 | W | 0x00000000 | System control1 |
| CSITX SYS CTRL2 | 0x0018 | W | 0x00000001 | System control2 |
| CSITX SYS CTRL3 IMD | 0x001c | W | 0x00000001 | System control3 |
| CSITX TIMING CTRL | 0x0020 | W | 0x00000000 | System timing control |
| CSITX TIMING VPW NUM | 0x0024 | W | 0x00000000 | VPW number |
| CSITX TIMING VBP NUM | 0x0028 | W | 0x00000000 | VBP number |
| CSITX TIMING VFP NUM | 0x002c | W | 0x00000000 | VFP number |
| CSITX TIMING HPW PAD DING NUM | 0x0030 | W | 0x00000000 | HPW number in padding mode |
| CSITX VOP PATH CTRL | 0x0040 | W | 0x00002a0f | VOP path control |
| CSITX VOP PATH PKT CTR | 0x0050 | W | 0x00000000 | VOP path packet control |
| CSITX BYPASS CTRL | 0x0060 | W | 0x06406b08 | Bypass path control |
| CSITX BYPASS PKT CTR | 0x0064 | W | 0x00000000 | Bypass path packet control |
| CSITX BYPASS D GAIN CTRL | 0x0068 | W | 0x00000000 | D_gain control |
| CSITX BYPASS D GAIN ID | 0x006c | W | 0x00000000 | D_gain ID |
| CSITX CSITX STATUS0 | 0x0070 | W | 0x00000000 | CSITX status0 |
| CSITX CSITX STATUS1 | 0x0074 | W | 0x00000000 | CSITX status1 |
| CSITX CSITX STATUS2 | 0x0078 | W | 0x00000000 | CSITX status2 |
| CSITX LINE FLAG NUM | 0x007c | W | 0x00000000 | Line flag number |
| CSITX INTR EN IMD | 0x0080 | W | 0x00000000 | Interrupt enable |
| CSITX INTR CLR IMD | 0x0084 | W | 0x00000000 | Interrupt clear |
| CSITX INTR STATUS | 0x0088 | W | 0x00000000 | Interrupt status |
| CSITX INTR RAW STATUS | 0x008c | W | 0x00000000 | Interrupt raw status |
| CSITX ERR INTR EN IMD | 0x0090 | W | 0x00000000 | Error interrupt enable |
| CSITX ERR INTR CLR IMD | 0x0094 | W | 0x00000000 | Error interrupt enable |
| CSITX ERR INTR STATUS IMD | 0x0098 | W | 0x00000000 | Error interrupt enable |
| CSITX ERR INTR RAW STATUS IMD | 0x009c | W | 0x00000000 | Error interrupt enable |
| CSITX ULPS CTRL IMD | 0x00a0 | W | 0x00000000 | ULPS mode control |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------|--------|------|-------------|----------------------|
| CSITX_LPDT_CTRL_IMD | 0x00a4 | W | 0x00000000 | LPDT mode control |
| CSITX_LPDT_DATA_IMD | 0x00a8 | W | 0x00000000 | LPDT data |
| CSITX_DPHY_CTRL | 0x00b0 | W | 0x00000078 | DPHY control |
| CSITX_BYPASS_DPCC_CTRL | 0x00e0 | W | 0x00000000 | DPCC control |
| CSITX_BYPASS_DPCC_THRESHOLD | 0x00e4 | W | 0x00000000 | DPCC threshold value |
| CSITX_BYPASS_DPCC_OFFSET | 0x00e8 | W | 0x00000000 | DPCC offset |
| CSITX_BYPASS_DPCC_REGION_NUMBER | 0x00ec | W | 0x00000000 | DPCC region number |
| CSITX_BYPASS_DPCC_POINT0 | 0x00f0 | W | 0x00000000 | DPCC point0 |
| CSITX_BYPASS_DPCC_POINT1 | 0x00f4 | W | 0x00000000 | DPCC point1 |
| CSITX_BYPASS_DPCC_POINT2 | 0x00f8 | W | 0x00000000 | DPCC point2 |
| CSITX_BYPASS_DPCC_POINT3 | 0x00fc | W | 0x00000000 | DPCC point3 |
| CSITX_BYPASS_DPCC_POINT4 | 0x0100 | W | 0x00000000 | DPCC point4 |
| CSITX_BYPASS_DPCC_POINT5 | 0x0104 | W | 0x00000000 | DPCC point5 |
| CSITX_BYPASS_DPCC_POINT6 | 0x0108 | W | 0x00000000 | DPCC point6 |
| CSITX_BYPASS_DPCC_POINT7 | 0x010c | W | 0x00000000 | DPCC point7 |
| CSITX_BYPASS_DPCC_POINT8 | 0x0110 | W | 0x00000000 | DPCC point8 |
| CSITX_BYPASS_DPCC_POINT9 | 0x0114 | W | 0x00000000 | DPCC point9 |
| CSITX_BYPASS_DPCC_POINT10 | 0x0118 | W | 0x00000000 | DPCC point10 |
| CSITX_BYPASS_DPCC_POINT11 | 0x011c | W | 0x00000000 | DPCC point11 |
| CSITX_BYPASS_DPCC_POINT12 | 0x0120 | W | 0x00000000 | DPCC point12 |
| CSITX_BYPASS_DPCC_POINT13 | 0x0124 | W | 0x00000000 | DPCC point13 |
| CSITX_BYPASS_DPCC_POINT14 | 0x0128 | W | 0x00000000 | DPCC point14 |
| CSITX_BYPASS_DPCC_POINT15 | 0x012c | W | 0x00000000 | DPCC point15 |

Notes:B- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

13.4.3 Detail Register Description

CSITX CONFIG DONE

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|---------|-------------|--|
| 31:9 | RO | 0x0 | Reserved |
| 8 | RW | 0x0 | config_done_mode 1'b0 : Config done when frm_end_rx 1'b1 : Config done when frm_end_tx |
| 7:5 | RO | 0x0 | Reserved |
| 4 | W1 C | 0x0 | config_done_imd Make register valid immediately |
| 3:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | config_done In the first setting of the register, the new value was saved into the mirror register. When all the register config finish, writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame |

CSITX CSITX EN

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---|
| 31 | RW | 0x0 | gating_en Auto gating 1'b0 : Disable 1'b1 : Enable |
| 30:23 | RO | 0x0 | Reserved |
| 22 | RW | 0x0 | gating_idi_clk_dis 1'b0 : Disable 1'b1 : Enable |
| 21 | RW | 0x0 | gating_vop_clk_dis 1'b0 : Disable 1'b1 : Enable |
| 20 | RW | 0x0 | gating_cam_clk_dis 1'b0 : Disable 1'b1 : Enable |
| 19 | RW | 0x0 | gating_esc_clk_dis 1'b0 : Disable 1'b1 : Enable |
| 18 | RW | 0x0 | gating_hs_clk_dis 1'b0 : Disable 1'b1 : Enable |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 17 | RW | 0x0 | gating_wordhs_clk_dis 1'b0 : Disable 1'b1 : Enable |
| 16 | RW | 0x0 | gating_bytehs_clk_dis 1'b0 : Disable 1'b1 : Enable |
| 15:10 | RO | 0x0 | Reserved |
| 9 | RW | 0x0 | idi_48bit_en 1'b0 : Disable 1'b1 : Enable |
| 8 | RW | 0x0 | vop_p2_en 1'b0 : Disable 1'b1 : Enable |
| 7:6 | RO | 0x0 | Reserved |
| 5:4 | RW | 0x0 | lane_num 2'b00 : One-lane 2'b01 : Two-lane 2'b10 : Three-lane 2'b11 : Four-lane |
| 3 | RO | 0x0 | Reserved |
| 2 | RW | 0x0 | dphy_en 1'b0 : Disable 1'b1 : Enable |
| 0 | RW | 0x0 | csitx_en 1'b0 : Disable 1'b1 : Enable |

CSITX CSITX VERSION

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------|
| 31:0 | RW | 0x00000000 | csitx_version CSITX version |

CSITX SYS CTRL0 IMD

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | soft_rst Soft reset 1'b0 : Disable 1'b1 : Enable |

CSITX SYS CTRL1

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | bypass_select 1'b0 : Select vop 1'b1 : Select idi |

CSITX_SYS_CTRL2

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:25 | RO | 0x0 | Reserved |
| 24:16 | RW | 0x000 | adv_read_pld_num If adv_read_pld_en is 1 , start to read pld fifo when pld_fifo_space2empty >= wc_128b - adv_read_pld_num |
| 15:9 | RO | 0x0 | Reserved |
| 8 | RW | 0x0 | adv_read_pld_en 1'b0 : Disable 1'b1 : Enable |
| 7:6 | RO | 0x0 | Reserved |
| 5 | RW | 0x0 | vop_whole_frm_en 1'b0 : Disable 1'b1 : Enable |
| 4 | RW | 0x0 | idi_whole_frm_en 1'b0 : Disable 1'b1 : Enable |
| 3:2 | RO | 0x0 | Reserved |
| 1 | RW | 0x0 | hsync_enable Send hsync timing short packet |
| 0 | RW | 0x1 | vsync_enable Send vsync timing short packet |

CSITX_SYS_CTRL3 IMD

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:9 | RO | 0x0 | Reserved |
| 8 | RW | 0x0 | cont_mode_clk_clr Disable clock lane txrequesths in continous mode |
| 7:5 | RO | 0x0 | Reserved |
| 4 | RW | 0x0 | cont_mode_clk_set Enable clock lane txrequesths in continous mode |
| 3:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x1 | non_continious_mode Non-continous mode |

CSITX_TIMING_CTRL

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | hpw_cnt_num HPW timing adjustment number |
| 15:5 | RO | 0x0 | Reserved |
| 4 | RW | 0x0 | hpw_padding_en Enable hpw timing adjustment in padding mode 1'b0 : Disable 1'b1 : Enable |
| 3 | RW | 0x0 | hpw_en Enable hpw timing adjustment 1'b0 : Disable 1'b1 : Enable |
| 2 | RW | 0x0 | vfp_en Enable vfp timing adjustment 1'b0 : Disable 1'b1 : Enable |
| 1 | RW | 0x0 | vbp_en Enable vbp timing adjustment 1'b0 : Disable 1'b1 : Enable |
| 0 | RW | 0x0 | vpw_en Enable vpw timing adjustment 1'b0 : Disable 1'b1 : Enable |

CSITX TIMING VPW NUM

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | vpw_cnt_num VPW timing adjustment number |

CSITX TIMING VBP NUM

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | vbp_cnt_num VBP timing adjustment number |

CSITX TIMING VFP NUM

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:0 | RW | 0x00000000 | vfp_cnt_num VFP timing adjustment number |

CSITX TIMING HPW PADDING NUM

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RO | 0x0 | Reserved |
| 15:0 | RW | 0x0000 | hpw_padding_num HPW timing adjustment number in padding mode |

CSITX_VOP_PATH_CTRL

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | vop_wc_userdefine Word count userdefine |
| 15:14 | RW | 0x0 | vop_vc_userdefine Virtual num userdefine |
| 13:8 | RW | 0x2a | vop_dt_userdefine Datatype userdefine |
| 7:4 | RW | 0x0 | vop_pixel_format 4'b0000 : Raw8 4'b0001 : Raw10 4'b0010 : Pixel10 4'b0011 : Pixel128 4'b0100 : Raw12 4'b0101 : Pixel12 4'b1xxx : Reserved |
| 3 | RW | 0x1 | vop_wc_userdefine_en Enable word count userdefine 1'b0 : Disable 1'b1 : Enable |
| 2 | RW | 0x1 | vop_vc_userdefine_en Enable virtual num userdefine 1'b0 : Disable 1'b1 : Enable |
| 1 | RW | 0x1 | vop_dt_userdefine_en Enable datatype userdefine 1'b0 : Disable 1'b1 : Enable |
| 0 | RW | 0x0 | vop_path_en Enable channel0 1'b0 : Disable 1'b1 : Enable |

CSITX_VOP_PATH_PKT_CTRL

Address: Operational Base + offset (0x0050)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | vop_wc_active The active byte number of one packet |
| 15:9 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 8 | RW | 0x0 | vop_pkt_padding_en Enable packet padding 1'b0 : Disable 1'b1 : Enable |
| 7:5 | RW | 0x0 | vop_line_padding_num Line padding number , real - 1 |
| 4 | RW | 0x0 | vop_line_padding_en Enable packet padding 1'b0 : Disable 1'b1 : Enable |
| 3:0 | RO | 0x0 | Reserved |

CSITX BYPASS CTRL

Address: Operational Base + offset (0x0060)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | bypass_wc_userdefine Word count userdefine |
| 15:14 | RW | 0x1 | bypass_vc_userdefine Virtual num userdefine |
| 13:8 | RW | 0x2b | bypass_dt_userdefine Datatype userdefine |
| 7:4 | RW | 0x0 | bypass_cam_format 4'b0000 : Raw8 4'b0001 : Raw10 4'b0010 : Pixel10 4'b0011 : Pixel128(used) 4'b11xx : Reserved |
| 3 | RW | 0x1 | bypass_wc_userdefine_en Enable word count userdefine 1'b0 : Disable 1'b1 : Enable |
| 2 | RW | 0x0 | bypass_vc_userdefine_en Enable virtual num userdefine 1'b0 : Disable 1'b1 : Enable |
| 1 | RW | 0x0 | bypass_dt_userdefine_en Enable datatype userdefine 1'b0 : Disable 1'b1 : Enable |
| 0 | RW | 0x0 | bypass_path_en Enable bypass path 1'b0 : Disable 1'b1 : Enable |

CSITX BYPASS PKT CTRL

Address: Operational Base + offset (0x0064)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:16 | RW | 0x0000 | bypass_wc_active The active byte number of one packet |
| 15:9 | RO | 0x0 | Reserved |
| 8 | RW | 0x0 | bypass_pkt_padding_en Enable packet padding 1'b0 : Disable 1'b1 : Enable |
| 7:5 | RW | 0x0 | bypass_line_padding_num Line padding number , real - 1 |
| 4 | RW | 0x0 | bypass_line_padding_en Enable packet padding 1'b0 : Disable 1'b1 : Enable |
| 3:0 | RO | 0x0 | Reserved |

CSITX BYPASS D GAIN CTRL

Address: Operational Base + offset (0x0068)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:13 | RW | 0x000000 | d_gain D_gain value = (1~256)*2^10 |
| 12:3 | RW | 0x000 | black_level Black level |
| 2:1 | RW | 0x0 | d_gain_format 2'b00 : Raw8 2'b01 : Raw10 2;b1x : Raw12 |
| 0 | RW | 0x0 | d_gain_en 1'b0 : Disable 1'b1 : Enable |

CSITX BYPASS D GAIN ID

Address: Operational Base + offset (0x006c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------|
| 31:24 | RW | 0x00 | d_gain_id3 D_gain ID3 |
| 23:16 | RW | 0x00 | d_gain_id2 D_gain ID2 |
| 15:8 | RW | 0x00 | d_gain_id1 D_gain ID1 |
| 7:0 | RW | 0x00 | d_gain_id0 D_gain ID0 |

CSITX CSITX STATUS0

Address: Operational Base + offset (0x0070)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | lane_splitter_idle Lane splitter idle |
| 5 | RO | 0x0 | packetiser_idle Packetizer idle |
| 4 | RO | 0x0 | tx_buff_empty TX fifo empty |
| 3 | RO | 0x0 | pld_lb_valid Payload lb valid |
| 2 | RO | 0x0 | pld_lb_empty Payload lb empty |
| 1 | RO | 0x0 | hdr_fifo_empty Header fifo empty |
| 0 | RO | 0x0 | csitx_idle CSITX idle |

CSITX CSITX STATUS1

Address: Operational Base + offset (0x0074)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:20 | RO | 0x0 | Reserved |
| 19 | RW | 0x0 | txreadyesc_3 Date lane3 txreadyesc |
| 18 | RO | 0x0 | txreadyesc_2 Date lane2 txreadyesc |
| 17 | RO | 0x0 | txreadyesc_1 Date lane1 txreadyesc |
| 16 | RO | 0x0 | txreadyesc_0 Date lane0 txreadyesc |
| 15 | RO | 0x0 | txulpsactivenot_3 Date lane3 txulpsactivenot |
| 14 | RO | 0x0 | txulpsactivenot_2 Date lane2 txulpsactivenot |
| 13 | RO | 0x0 | txulpsactivenot_1 Date lane1 txulpsactivenot |
| 12 | RO | 0x0 | txulpsactivenot_0 Date lane0 txulpsactivenot |
| 11 | RO | 0x0 | txreadyhs_3 Lane3 txreadyhs |
| 10 | RO | 0x0 | txreadyhs_2 Lane2 txreadyhs |
| 9 | RO | 0x0 | txreadyhs_1 Lane1 txreadyhs |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 8 | RO | 0x0 | txreadyhs_0 Lane0 txreadyhs |
| 7 | RO | 0x0 | stopstate_3 Lane3 stopstate |
| 6 | RO | 0x0 | stopstate_2 Lane2 stopstate |
| 5 | RO | 0x0 | stopstate_1 Lane1 stopstate |
| 4 | RO | 0x0 | stopstate_0 Lane0 stopstate |
| 3 | RO | 0x0 | Reserved |
| 2 | RO | 0x0 | txulpsactivenot_clk CLK lane txulpsactivenot |
| 1 | RO | 0x0 | stopstate_clk CLK lane lock |
| 0 | RO | 0x0 | dphy_pll_lock DPHY PLL lock |

CSITX CSITX STATUS2

Address: Operational Base + offset (0x0078)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------|
| 31:16 | RW | 0x0000 | cphy_scanline CPHY scan line |
| 15:0 | RO | 0x0000 | dphy_scanline DPHY scan line |

CSITX LINE FLAG NUM

Address: Operational Base + offset (0x007c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------------------|
| 31:16 | RW | 0x0000 | line_num_flag1 Line num flag1 |
| 15:0 | RW | 0x0000 | line_num_flag0 Line num flag0 |

CSITX INTR EN IMD

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | Reserved |
| 10 | RW | 0x0 | intr_en_csitx_idle Enable csitx idle interrupt |
| 9 | RW | 0x0 | intr_en_pll_lock Enable pll lock interrupt |
| 8 | RW | 0x0 | intr_en_stopstate Enable dphy stopstate interrupt |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7 | RW | 0x0 | intr_en_line_flag1 Enable line_flag1 interrupt |
| 6 | RW | 0x0 | intr_en_line_flag0 Enable line_flag0 interrupt |
| 5 | RW | 0x0 | intr_en_line_end_tx Enable transmitted line_end interrupt |
| 4 | RW | 0x0 | intr_en_frm_end_tx Enable transmitted frm_end interrupt |
| 3 | RW | 0x0 | intr_en_frm_st_tx Enable transmitted frm_st interrupt |
| 2 | RW | 0x0 | intr_en_line_end_rx Enable received line_end interrupt |
| 1 | RW | 0x0 | intr_en_frm_end_rx Enable received frm_end interrupt |
| 0 | RW | 0x0 | intr_en_frm_st_rx Enable received frm_st interrupt |

CSITX INTR CLR IMD

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | Reserved |
| 10 | W1C | 0x0 | intr_clr_csitx_idle Clear csitx idle interrupt |
| 9 | W1C | 0x0 | intr_clr_pll_lock Clear pll lock interrupt |
| 8 | W1C | 0x0 | intr_clr_stopstate Clear dphy stopstate interrupt |
| 7 | W1C | 0x0 | intr_clr_line_flag1 Clear line_flag1 interrupt |
| 6 | W1C | 0x0 | intr_clr_line_flag0 Clear line_flag0 interrupt |
| 5 | W1C | 0x0 | intr_clr_line_end_tx Clear transmitted line_end interrupt |
| 4 | W1C | 0x0 | intr_clr_frm_end_tx Clear transmitted frm_end interrupt |
| 3 | W1C | 0x0 | intr_clr_frm_st_tx Clear transmitted frm_st interrupt |
| 2 | W1C | 0x0 | intr_clr_line_end_rx Clear received line_end interrupt |
| 1 | W1C | 0x0 | intr_clr_frm_end_rx Clear received frm_end interrupt |
| 0 | W1C | 0x0 | intr_clr_frm_st_rx Clear received frm_st interrupt |

CSITX INTR STATUS

Address: Operational Base + offset (0x0088)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | Reserved |
| 10 | W1 C | 0x0 | intr_sts_csitx_idle CSITX idle interrupt status |
| 9 | RO | 0x0 | intr_sts_pll_lock PHY PLL lock interrupt status |
| 8 | RO | 0x0 | intr_sts_stopstate PHY stopstate interrupt status |
| 7 | RO | 0x0 | intr_sts_line_flag1 Line flag1 interrupt status |
| 6 | RO | 0x0 | intr_sts_line_flag0 Line flag0 interrupt status |
| 5 | RO | 0x0 | intr_sts_frame_end_tx TX frame end interrupt status |
| 4 | RO | 0x0 | intr_sts_line_end_tx TX line end interrupt status |
| 3 | RO | 0x0 | intr_sts_frm_st_tx TX frame start interrupt status |
| 2 | RO | 0x0 | intr_sts_frm_end_rx RX frame end interrupt status |
| 1 | RO | 0x0 | intr_sts_line_end_rx RX line end interrupt status |
| 0 | RW | 0x0 | intr_sts_frm_st_rx RX frame start interrupt status |

CSITX INTR RAW STATUS

Address: Operational Base + offset (0x008c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:11 | RO | 0x0 | Reserved |
| 10 | RW | 0x0 | intr_csitx_idle CSITX idle interrupt raw status |
| 9 | RO | 0x0 | intr_pll_lock PHY PLL lock interrupt raw status |
| 8 | RO | 0x0 | intr_stopstate PHY stopstate interrupt raw status |
| 7 | RO | 0x0 | intr_line_flag1 Line flag1 interrupt raw status |
| 6 | RO | 0x0 | intr_line_flag0 Line flag0 interrupt raw status |
| 5 | RO | 0x0 | intr_frm_end_tx TX frame end interrupt raw status |
| 4 | RO | 0x0 | intr_line_end_tx TX line end interrupt raw status |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 3 | RO | 0x0 | intr_frm_st_tx TX frame start interrupt raw status |
| 2 | RO | 0x0 | intr_line_end_rx RX frame end interrupt raw status |
| 1 | RO | 0x0 | intr_frm_end_rx RX line end interrupt raw status |
| 0 | RO | 0x0 | intr_frm_st_rx RX frame start interrupt raw status |

CSITX_ERR_INTR_EN_IMD

Address: Operational Base + offset (0x0090)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | Reserved |
| 11 | RW | 0x0 | intr_en_txreadyhs_error1 Enable txreadyhs error1 interrupt |
| 10 | RW | 0x0 | intr_en_txreadyhs_error0 Enable txreadyhs error0 interrupt |
| 9 | RW | 0x0 | intr_en_outbuffer_underflow Enable output fifo underflow interrupt |
| 8 | RW | 0x0 | intr_en_outbuffer_overflow Enable output fifo overflow interrupt |
| 7 | RW | 0x0 | intr_en_pld_fifo_underflow Enable payload fifo underflow interrupt |
| 6 | RW | 0x0 | intr_en_pld_fifo_overflow Enable payload fifo overflow interrupt |
| 5 | RW | 0x0 | intr_en_hdr_fifo_underflow Enable header fifo underflow interrupt |
| 4 | RW | 0x0 | intr_en_hdr_fifo_overflow Enable header fifo overflow interrupt |
| 3 | RW | 0x0 | intr_en_idi_pld_fifo_underflow Enable idi payload fifo underflow interrupt |
| 2 | RW | 0x0 | intr_en_idi_pld_fifo_overflow Enable idi payload fifo overflow interrupt |
| 1 | RW | 0x0 | intr_en_idi_hdr_fifo_underflow Enable idi header fifo underflow interrupt |
| 0 | RW | 0x0 | intr_en_idi_hdr_fifo_overflow Enable idi header fifo overflow interrupt |

CSITX_ERR_INTR_CLR_IMD

Address: Operational Base + offset (0x0094)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | Reserved |
| 11 | RW | 0x0 | intr_clr_txreadyhs_error1 Clear txreadyhs error1 interrupt |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 10 | RW | 0x0 | intr_clr_txreadyhs_error0 Clear txreadyhs error0 interrupt |
| 9 | RW | 0x0 | intr_clr_outbuffer_underflow Clear output fifo underflow interrupt |
| 8 | RW | 0x0 | intr_clr_outbuffer_overflow Clear output fifo overflow interrupt |
| 7 | RW | 0x0 | intr_clr_pld_fifo_underflow Clear payload fifo underflow interrupt |
| 6 | RO | 0x0 | Reserved |
| 5 | RW | 0x0 | intr_clr_hdr_fifo_underflow Clear header fifo underflow interrupt |
| 4 | RW | 0x0 | intr_clr_hdr_fifo_overflow Clear header fifo overflow interrupt |
| 3 | RW | 0x0 | intr_clr_idi_pld_fifo_underflow Clear idi payload fifo underflow interrupt |
| 2 | RW | 0x0 | intr_clr_idi_pld_fifo_overflow Clear idi payload fifo overflow interrupt |
| 1 | RW | 0x0 | intr_clr_idi_hdr_fifo_underflow Clear idi header fifo underflow interrupt |
| 0 | RW | 0x0 | intr_clr_idi_hdr_fifo_overflow Clear idi header fifo overflow interrupt |

CSITX_ERR_INTR_STATUS_IMD

Address: Operational Base + offset (0x0098)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | Reserved |
| 11 | RW | 0x0 | intr_sts_txreadyhs_error1 Txreadyhs error1 interrupt |
| 10 | RW | 0x0 | intr_sts_txreadyhs_error0 Txreadyhs error0 interrupt |
| 9 | RW | 0x0 | intr_sts_outbuffer_underflow Output fifo underflow interrupt |
| 8 | RW | 0x0 | intr_sts_outbuffer_overflow Output fifo overflow interrupt |
| 7 | RW | 0x0 | intr_sts_pld_fifo_underflow Payload fifo underflow interrupt |
| 6 | RW | 0x0 | intr_sts_pld_fifo_overflow Payload fifo overflow interrupt |
| 5 | RW | 0x0 | intr_sts_hdr_fifo_underflow Header fifo underflow interrupt |
| 4 | RW | 0x0 | intr_sts_hdr_fifo_overflow Header fifo overflow interrupt |
| 3 | RW | 0x0 | intr_sts_idi_pld_fifo_underflow IDI payload fifo underflow interrupt |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 2 | RW | 0x0 | intr_sts_idi_pld_fifo_overflow IDI payload fifo overflow interrupt |
| 1 | RW | 0x0 | intr_sts_idi_hdr_fifo_underflow IDI header fifo underflow interrupt |
| 0 | RW | 0x0 | intr_sts_idi_hdr_fifo_overflow IDI header fifo overflow interrupt |

CSITX_ERR_INTR_RAW_STATUS_IMD

Address: Operational Base + offset (0x009c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | Reserved |
| 11 | RW | 0x0 | intr_txreadyhs_error1 Txreadyhs error1 raw interrupt |
| 10 | RW | 0x0 | intr_txreadyhs_error0 Txreadyhs error0 raw interrupt |
| 9 | RW | 0x0 | intr_outbuffer_underflow Output fifo underflow raw interrupt |
| 8 | RW | 0x0 | intr_outbuffer_overflow Output fifo overflow raw interrupt |
| 7 | RW | 0x0 | intr_pld_fifo_underflow Payload fifo underflow raw interrupt |
| 6 | RW | 0x0 | intr_pld_fifo_overflow Payload fifo overflow raw interrupt |
| 5 | RW | 0x0 | intr_hdr_fifo_underflow Header fifo underflow raw interrupt |
| 4 | RW | 0x0 | intr_hdr_fifo_overflow Header fifo overflow raw interrupt |
| 3 | RW | 0x0 | intr_idi_pld_fifo_underflow IDI payload fifo underflow raw interrupt |
| 2 | RW | 0x0 | intr_idi_pld_fifo_overflow IDI payload fifo overflow raw interrupt |
| 1 | RW | 0x0 | intr_idi_hdr_fifo_underflow IDI header fifo underflow raw interrupt |
| 0 | RW | 0x0 | intr_idi_hdr_fifo_overflow IDI header fifo overflow raw interrupt |

CSITX_ULPS_CTRL_IMD

Address: Operational Base + offset (0x00a0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:16 | RW | 0x0000 | ulps_exit_wait The waiting time of exiting ulps mode |
| 15:12 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 11 | RW | 0x0 | data3_lane_ulps_exit Disable data3 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 10 | RW | 0x0 | data3_lane_ulps_entry Enable data3 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 9 | RW | 0x0 | data2_lane_ulps_exit Disable data2 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 8 | RW | 0x0 | data2_lane_ulps_entry Enable data2 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 7 | RW | 0x0 | data1_lane_ulps_exit Disable data1 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 6 | RW | 0x0 | data1_lane_ulps_entry Enable data1 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 5 | RW | 0x0 | data0_lane_ulps_exit Disable data0 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 4 | RW | 0x0 | data0_lane_ulps_entry Enable data0 lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 3 | RW | 0x0 | clk_lane_ulps_exit Disable clk lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 2 | RW | 0x0 | clk_lane_ulps_entry Enable clk lane ulps mode 1'b0 : Disable 1'b1 : Enable |
| 1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | ulps_en Enable ulps mode 1'b0 : Disable 1'b1 : Enable |

CSITX LPDT CTRL IMD

Address: Operational Base + offset (0x00a4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RO | 0x0 | Reserved |
| 3:2 | RW | 0x0 | lpdt_lane_num LPDT lane number 2'b00 : One lane 2'b01 : Two lane 2'b1x : Four lane |
| 1 | RW | 0x0 | lpdt_start Start LPDT transfer |
| 0 | RW | 0x0 | lpdt_en Enable LPDT mode |

CSITX LPDT DATA IMD

Address: Operational Base + offset (0x00a8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------|
| 31:0 | RW | 0x00000000 | lpdt_data LPDT data |

CSITX DPHY CTRL

Address: Operational Base + offset (0x00b0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | dphy_enable_3 Enable high speed data3 lane |
| 6 | RW | 0x1 | dphy_enable_2 Enable high speed data2 lane |
| 5 | RW | 0x1 | dphy_enable_1 Enable high speed data1 lane |
| 4 | RW | 0x1 | dphy_enable_0 Enable high speed data0 lane |
| 3 | RW | 0x1 | dphy_enableclk Enable high speed clock lane |
| 2:0 | RO | 0x0 | Reserved |

CSITX BYPASS DPCC CTRL

Address: Operational Base + offset (0x00e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | dpcc_point15_en 1'b0 : Disable 1'b1 : Enable |
| 30 | RW | 0x0 | dpcc_point14_en 1'b0 : Disable 1'b1 : Enable |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 29 | RW | 0x0 | dpcc_point13_en 1'b0 : Disable 1'b1 : Enable |
| 28 | RW | 0x0 | dpcc_point12_en 1'b0 : Disable 1'b1 : Enable |
| 27 | RW | 0x0 | dpcc_point11_en 1'b0 : Disable 1'b1 : Enable |
| 26 | RW | 0x0 | dpcc_point10_en 1'b0 : Disable 1'b1 : Enable |
| 25 | RW | 0x0 | dpcc_point9_en 1'b0 : Disable 1'b1 : Enable |
| 24 | RW | 0x0 | dpcc_point8_en 1'b0 : Disable 1'b1 : Enable |
| 23 | RW | 0x0 | dpcc_point7_en 1'b0 : Disable 1'b1 : Enable |
| 22 | RW | 0x0 | dpcc_point6_en 1'b0 : Disable 1'b1 : Enable |
| 21 | RW | 0x0 | dpcc_point5_en 1'b0 : Disable 1'b1 : Enable |
| 20 | RW | 0x0 | dpcc_point4_en 1'b0 : Disable 1'b1 : Enable |
| 19 | RW | 0x0 | dpcc_point3_en 1'b0 : Disable 1'b1 : Enable |
| 18 | RW | 0x0 | dpcc_point2_en 1'b0 : Disable 1'b1 : Enable |
| 17 | RW | 0x0 | dpcc_point1_en 1'b0 : Disable 1'b1 : Enable |
| 16 | RW | 0x0 | dpcc_point0_en 1'b0 : Disable 1'b1 : Enable |
| 15:14 | RO | 0x0 | Reserved |
| 13:8 | RW | 0x00 | dpcc_mul DPCC multiplier |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | dpcc_mode DPCC mode 2'b00 : If the selected pixel \geq sw_dpcc_thold , it will be replaced by 12'h0 , otherwise replaced by 12'hfff 2'b01 : If the second pixel on the left \geq sw_dpcc_thold , it will be replaced by 12'h0 , otherwise replaced by 12'hfff 2'b10 : Replace the selected pixel with the second pixel on the left 2'b11 : Replace the selected pixel with the selected pixel * sw_dpcc_mul |
| 5:4 | RW | 0x0 | dpcc_fmt DPCC format 2'b00 : Raw8 2'b01 : Raw10 2'b1x : Raw12 |
| 3 | RO | 0x0 | Reserved |
| 2 | RW | 0x0 | dpcc_en 1'b0 : Disable 1'b1 : Enable |
| 1 | RW | 0x0 | bypass_dpcc_en 1'b0 : Disable 1'b1 : Enable |
| 0 | RW | 0x0 | bypass_dgain_en 1'b0 : Disable 1'b1 : Enable |

CSITX BYPASS DPCC THOLD

Address: Operational Base + offset (0x00e4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|------------------------------|
| 31:24 | RW | 0x00 | dpcc_wrapy Real - 1 |
| 23:16 | RW | 0x00 | dpcc_wrapx Real - 1 |
| 15:12 | RO | 0x0 | Reserved |
| 11:0 | RW | 0x000 | dpcc_thold DPCC threshold |

CSITX BYPASS DPCC OFFSET

Address: Operational Base + offset (0x00e8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------|
| 31:24 | RO | 0x0 | Reserved |
| 23:16 | RW | 0x00 | dpcc_offsety DPCC y offset |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------|
| 15:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0x00 | dpcc_offsetx DPCC x offset |

CSITX BYPASS DPCC WRAP NUM

Address: Operational Base + offset (0x00ec)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|----------------------------|
| 31:29 | RO | 0x0 | Reserved |
| 28:16 | RW | 0x0000 | dpcc_wrapy_num Real - 1 |
| 15:13 | RO | 0x0 | Reserved |
| 12:0 | RW | 0x0000 | dpcc_wrapx_num Real - 1 |

CSITX BYPASS DPCC POINT0

Address: Operational Base + offset (0x00f0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point0_y DPCC point0 y-axis |
| 7:0 | RW | 0x00 | dpcc_point0_x DPCC point0 x-axis |

CSITX BYPASS DPCC POINT1

Address: Operational Base + offset (0x00f4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point1_y DPCC point1 y-axis |
| 7:0 | RW | 0x00 | dpcc_point1_x DPCC point1 x-axis |

CSITX BYPASS DPCC POINT2

Address: Operational Base + offset (0x00f8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point2_y DPCC point2 y-axis |
| 7:0 | RW | 0x00 | dpcc_point2_x DPCC point2 x-axis |

CSITX BYPASS DPCC POINT3

Address: Operational Base + offset (0x00fc)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point3_y DPCC point3 y-axis |
| 7:0 | RW | 0x00 | dpcc_point3_x DPCC point3 x-axis |

CSITX BYPASS DPCC POINT4

Address: Operational Base + offset (0x0100)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point4_y DPCC point4 y-axis |
| 7:0 | RW | 0x00 | dpcc_point4_x DPCC point4 x-axis |

CSITX BYPASS DPCC POINT5

Address: Operational Base + offset (0x0104)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point5_y DPCC point5 y-axis |
| 7:0 | RW | 0x00 | dpcc_point5_x DPCC point5 x-axis |

CSITX BYPASS DPCC POINT6

Address: Operational Base + offset (0x0108)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point6_y DPCC point6 y-axis |
| 7:0 | RW | 0x00 | dpcc_point6_x DPCC point6 x-axis |

CSITX BYPASS DPCC POINT7

Address: Operational Base + offset (0x010c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point7_y DPCC point7 y-axis |
| 7:0 | RW | 0x00 | dpcc_point7_x DPCC point7 x-axis |

CSITX BYPASS DPCC POINT8

Address: Operational Base + offset (0x0110)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point8_y DPCC point8 y-axis |
| 7:0 | RW | 0x00 | dpcc_point8_x DPCC point8 x-axis |

CSITX BYPASS DPCC POINT9

Address: Operational Base + offset (0x0114)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point9_y DPCC point9 y-axis |
| 7:0 | RW | 0x00 | dpcc_point9_x DPCC point9 x-axis |

CSITX BYPASS DPCC POINT10

Address: Operational Base + offset (0x0118)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point10_y DPCC point10 y-axis |
| 7:0 | RW | 0x00 | dpcc_point10_x DPCC point10 x-axis |

CSITX BYPASS DPCC POINT11

Address: Operational Base + offset (0x011c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point11_y DPCC point11 y-axis |
| 7:0 | RW | 0x00 | dpcc_point11_x DPCC point11 x-axis |

CSITX BYPASS DPCC POINT12

Address: Operational Base + offset (0x0120)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point12_y DPCC point12 y-axis |
| 7:0 | RW | 0x00 | dpcc_point12_x DPCC point12 x-axis |

CSITX BYPASS DPCC POINT13

Address: Operational Base + offset (0x0124)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point13_y DPCC point13 y-axis |
| 7:0 | RW | 0x00 | dpcc_point13_x DPCC point13 x-axis |

CSITX BYPASS DPCC POINT14

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point14_y DPCC point14 y-axis |
| 7:0 | RW | 0x00 | dpcc_point14_x DPCC point14 x-axis |

CSITX BYPASS DPCC POINT15

Address: Operational Base + offset (0x012c)

| Bit | Attr | Reset Value | Description |
|-------|------|-------------|---------------------------------------|
| 31:16 | RO | 0x0 | Reserved |
| 15:8 | RW | 0x00 | dpcc_point15_y DPCC point15 y-axis |
| 7:0 | RW | 0x00 | dpcc_point15_x DPCC point15 x-axis |

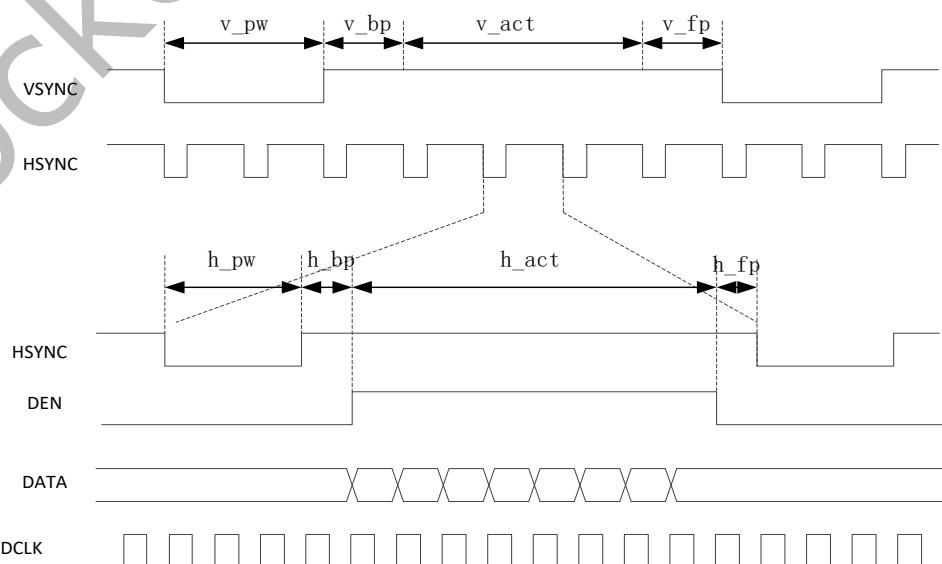
13.5 Interface Timing**13.5.1 Camera interface timing**

Fig. 13-3 Camera Interface Timing without pdaf

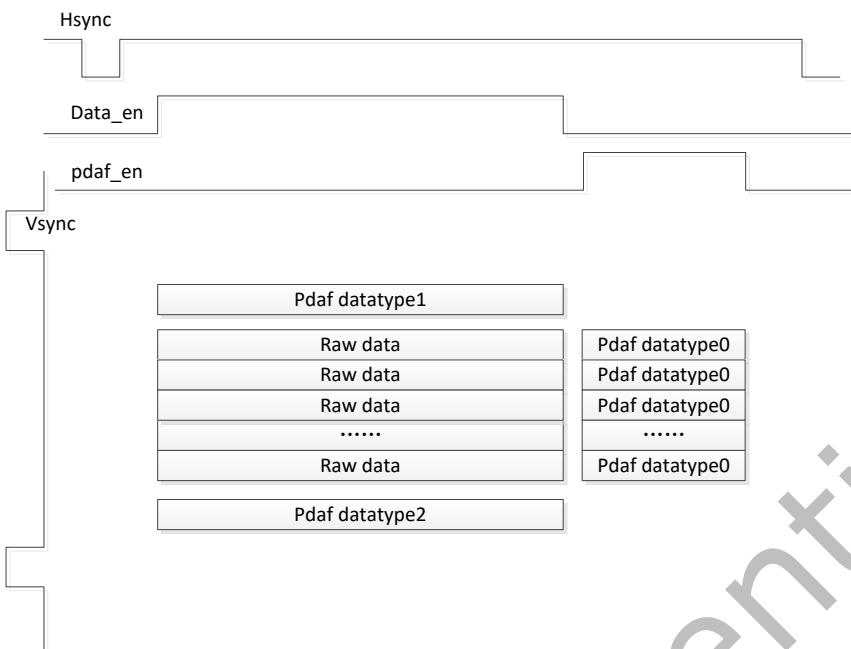


Fig. 13-4 Camera Interface Timing with pdaf

13.5.2 1.5.2 IDI interface timing

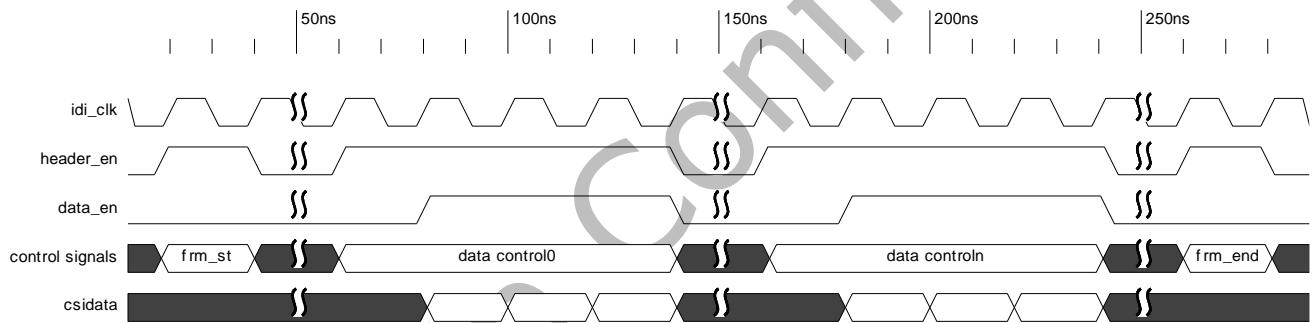


Fig. 13-5 IDI Interface Typical Timing

The `header_en` signal is used to indicate that new is being transferred at the Image Data Interface. It rises when a new packet becomes available at the interface and falls as soon as the packet finishes. The `data_en` signal is used to indicate that a new 32-bit word is available in `csi_data`, and it can only be set if `header_en` is also set. The `csi_data` is 32-bit and is transferred at each clock cycle. The `control_signals` are consist of 2-bit `virtual_channel`, 6-bit `data_type`, 16-bit `word_count` and 8-bit `ecc`.

13.5.3 1.5.3 APB interface timing

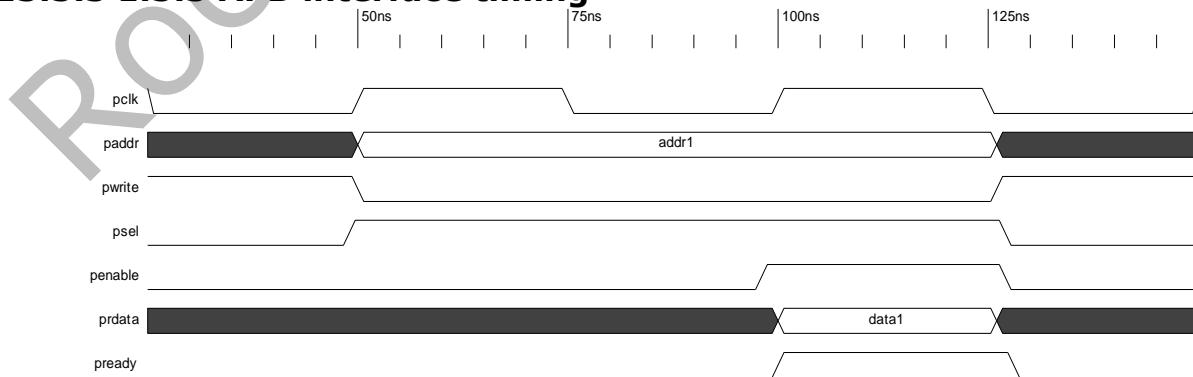


Fig. 13-6 APB Interface Typical Read Timing

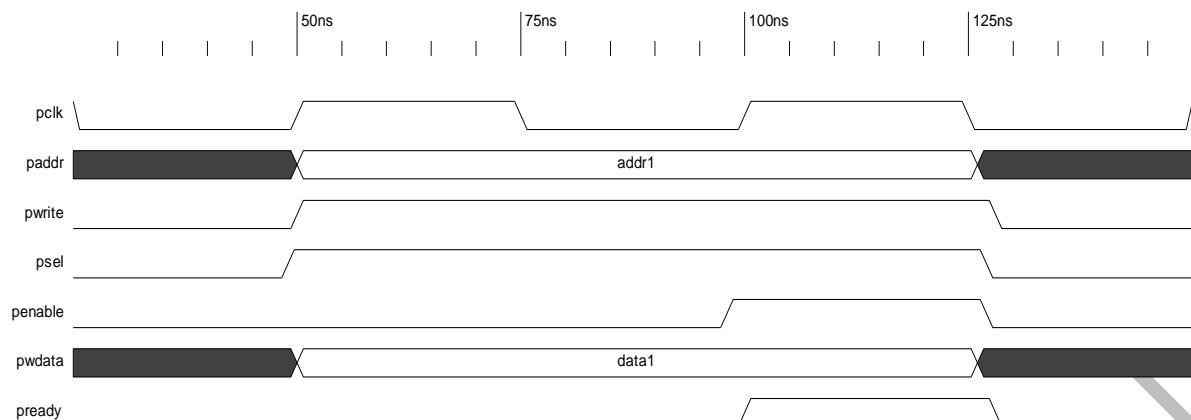


Fig. 13-7 APB Interface Typical Write Timing

13.5.4 DPHY interface timing

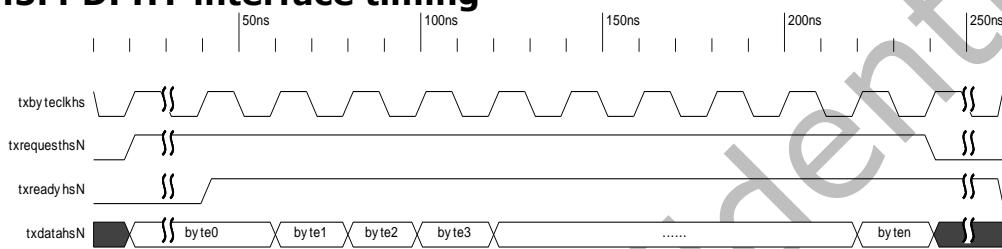


Fig. 13-8 DPHY Interface Typical Write Timing

TxdatahsN is an 8-bit data bus input of data laneN of DPHY that receives data to be transmitted. It is synchronous to the rising edge of txbyteclkhs. When txrequesths is sampled high, the DPHY lane module initiates a Start-of-Transmission(SOT) sequence . When txrequesths is sampled low while txreadyhs is asserted, the lane module initiates an End-of-Transmission(EOT) sequence. TxreadyhsN indicates that txdatahs is accepted by the lane module to be serially transmitted.

13.6 Application Notes

13.6.1 camera interface configuration guideline

CSI transmitter , to become operational , requires the following register to be programmed .

1. Program CSITX_EN to enable dphy_en , csitx_en and set lane number .
2. Send 0x1 to SYS_CTRL1 . Disable bypass_select .
3. Send 0x10 to SYS_CTRL3_IMD to enable continuous mode .Otherwise , send 0x1 to enable non-continuous mode .
4. Program VOP_PATH_CTRL according to application .
5. Program DPHY_CTRL to enable DPHY lane .
6. Send 0x10 to CONFIG_DONE .

The sequence below is for bypass mode .

1. Program CSITX_EN to enable dphy_en , csitx_en and set lane number .
2. Send 0x0 to SYS_CTRL1 . Enable bypass_select .
3. Send 0x10 to SYS_CTRL3_IMD to enable continuous mode .Otherwise , send 0x1 to enable non-continuous mode .
4. Program BYPASS_CTRL according to application .
5. Program DPHY_CTRL to enable DPHY lane .
6. Send 0x10 to CONFIG_DONE .

13.6.2 Switch between camera and IDI interface

When bypass_en is asserted , IDI interface is selected , otherwise camera interface is selected . Normally , to switch interface is based on frames .

Chapter 14 MIPI CSI-2 Host Controller

14.1 Overview

The CSI-2 Host Controller is designed to receive data from a CSI-2 compliant camera sensor. A DPHY configured as a Slave acts as the physical layer.

The MIPI CSI-2 Host Controller supports the following features:

- Compliant with MIPI Alliance Specification for CSI-2, Version 1.01.00-9 November 2010
- Interface with MIPI DPHY following PHY Protocol Interface, as defined in MIPI Alliance Specification for DPHY, Version 1.1-7 November 2011
- Up to four DPHY RX data lanes
- Dynamically configurable multi-lane merging
- Long and Short packet decoding
- Timing accurate signaling of Frame and Line synchronization packets
- Several Frame formats
 - General Frame or Digital Interlaced Video with or without accurate sync timing
 - Data Type (Packet or Frame Level) and Virtual Channel interleaving
- 32-bit Image Data Interface delivering data formatted as recommended in CSI-2 Specification
- All primary and secondary data formats
 - RGB, YUV, and RAW color space definitions
 - From 24-bit down to 6-bit per pixel
 - Generic or user-defined byte-based data types
- Error detection and correction
 - PHY level
 - Packet level
 - Line level
 - Frame level
- Support DSI video mode/command mode

14.2 Block Diagram

The following diagram shows the MIPI CSI-2 Host Controller architecture.

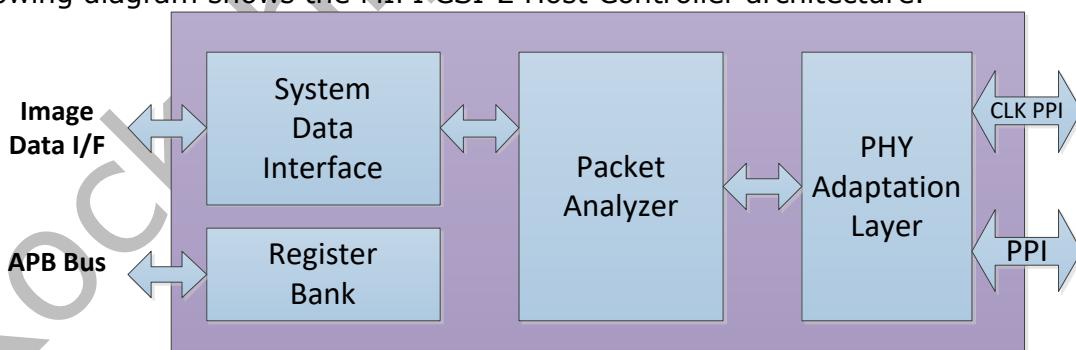


Fig. 14-1 MIPI CSI-2 Host Controller architecture

- **PHY Adaptation Layer:** Manages the DPHY PPI interface
- **Packet Analyzer:** Merges the data from the different lanes
- **Image Data Interface:** Reorders pixels into 32-bit data for memory storage and generates timing accurate video synchronization signals
- **AMBA-APB Register Bank:** Provides access to configuration and control registers

14.3 Function Description

14.3.1 Supported Resolutions and Frame Rates

The CSI-2 specification does not define the supported standard resolutions or frame rates. Camera sensor resolution, blanking periods, synchronization events, frame rates, and pixel color depth play a fundamental role in the required bandwidth. All these variables make it difficult to define a standard procedure to estimate the minimum lane rate and the minimum number of lanes that support a specific CSI-2 device.

Table 47-1 presents some predefined and supported camera settings, assuming the following:

- Clock lane frequency is 500 MHz or 750 MHz that results in a bandwidth of 1 Gbps or 1.5 Gbps respectively, for each data lane.
- No significant control/reserved traffic is present on the link when pixel data is being transmitted.

The last column of Table 47-1 presents the minimum number of lanes required for each configuration.

Table 14-1 Supported Camera Settings

| Mega Pixels | Mega Pixels with Overhead | Refresh Rate (Hz) | Color Depth (bpp) | CSI2 BW (Mbits) | DPHY at 1 Gbps Number of Lanes | DPHY at 1.5Gbps Number of Lanes |
|-------------------------|---------------------------|-------------------|-------------------|-----------------|--------------------------------|---------------------------------|
| 2MP | 2560000 | 15 | 24 | 922 | 1 | 1 |
| 2MP | 2560000 | 30 | 24 | 1843 | 2 | 2 |
| 3MP | 3840000 | 15 | 16 | 922 | 1 | 1 |
| 3MP | 3840000 | 30 | 16 | 1843 | 2 | 2 |
| 3MP | 3840000 | 30 | 24 | 2765 | 3 | 2 |
| 5MP | 6400000 | 15 | 16 | 1536 | 2 | 2 |
| 5MP | 6400000 | 15 | 24 | 2304 | 3 | 2 |
| 5MP | 6400000 | 30 | 16 | 3072 | 4 | 3 |
| 8MP | 10240000 | 15 | 16 | 2458 | 3 | 2 |
| 8MP | 10240000 | 15 | 24 | 3686 | 4 | 3 |
| 8MP | 10240000 | 30 | 12 | 3686 | 4 | 3 |
| 12MP | 15360000 | 15 | 12 | 2765 | 3 | 2 |
| 12MP | 15360000 | 15 | 16 | 3686 | 4 | 3 |
| 14MP | 17920000 | 15 | 12 | 3226 | 4 | 3 |
| 16MP | 20480000 | 15 | 12 | 3686 | 4 | 3 |
| Video Formats | | | | | | |
| 1280x720 pixels(720p) | 921600 | 30 | 24 | 664 | 1 | 1 |
| 1280x720 pixels(720p) | 921600 | 60 | 24 | 1327 | 2 | 1 |
| 1920x1080 pixels(1080p) | 2073600 | 60 | 24 | 2986 | 3 | 2 |

14.3.2 Error Detection

The CSI-2 Host Controller analyzes the received packets and determines if there are protocol errors. It is possible to monitor the following errors:

- Frame errors such as incorrect Frame sequence, reception of a CRC error in the most recent frame, and the mismatch between Frame Start and Frame End
- Line errors such as incorrect line sequence and mismatch between Line Start and Line End
- Packet errors such as ECC or CRC mismatch
- DPHY errors such as synchronization pattern mismatch

Table 47-2 shows all the errors that CSI-2 Host Controller can identify.

Table 14-2 Errors Identified by the CSI-2 Host Controller

| Error | Description | Level | Action |
|------------------------|---|-------|--|
| phy_errsotsynchs _* | Start of transmission error on data lane* | PHY | Packets with this error are not delivered in IDI |

| Error | Description | Level | Action |
|------------------------|---|--------|--|
| | with no synchronization achieved | | interface |
| phy_erresc_* | Escape entry error (ULPM) on data lane* | PHY | Informative only. Error is acknowledged in the register and the interrupt pin is raised. |
| phy_errsoths_* | Start of transmission error on data lane* but synchronization can still be achieved | PHY | Informative only since PHY can recover from this error. Error is acknowledged in register and the interrupt pin is raised. |
| vc*_err_crc | Checksum error detected on virtual channel* | Packet | Informative only. Error is acknowledged in the register and Interrupt pin is raised. |
| vc*_err_crc | Header ECC contains one error detected on virtual channel* | Packet | Informative only since controller can recover the correct header. Error is acknowledged in the register and the interrupt pin is raised. |
| err_ecc_double | Header ECC contains two errors. Unrecoverable. | Packet | Packets with this error are not delivered in IDI.s |
| err_id_vc* | Unrecognized or unimplemented data type detected in virtual channel* | Packet | Informative only. Error is acknowledged in the register and the interrupt pin is raised |
| err_f_bndry_matc h_vc* | Error matching Frame Start with Frame End for virtual channel* | Frame | Informative only. Error is acknowledged in register and the interrupt pin is raised if not masked. |
| err_f_seq_vc* | Incorrect Frame Sequence detected in virtual channel* | Frame | Informative only. Error is acknowledged in register and the interrupt pin is raised if not masked. |
| err_frame_data_vc* | Last received frame, in virtual channel*, had at least one CRC error | Frame | Informative only. Error is acknowledged in the register and the interrupt pin is raised. |

14.4 Register Description

14.4.1 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|------------------------|--------|------|-------------|-----------------------------------|
| CSI2HOST VERSION | 0x0000 | W | 0x00000000 | Controller version identification |
| CSI2HOST_N_LANES | 0x0004 | W | 0x00000000 | Number of active data lanes |
| CSI2HOST_dphy_shutdown | 0x0008 | W | 0x00000000 | DPHY shutdown control |
| CSI2HOST_DPHY_RSTZ | 0x000c | W | 0x00000000 | DPHY reset control |

| Name | Offset | Size | Reset Value | Description |
|------------------------|--------|------|-------------|--|
| CSI2HOST_CS1_RESETN | 0x0010 | W | 0x00000000 | CSI2 controller reset |
| CSI2HOST_PHY_STATE | 0x0014 | W | 0x00000000 | General settings for all blocks |
| CSI2HOST_DATA_IDS_1 | 0x0018 | W | 0x00000000 | Data IDS for which IDI reports line boundary matching errors |
| CSI2HOST_DATA_IDS_2 | 0x001c | W | 0x00000000 | Data IDS for which IDI reports line boundary matching errors |
| CSI2HOST_ERR1 | 0x0020 | W | 0x00000000 | Error state register 1 |
| CSI2HOST_ERR2 | 0x0024 | W | 0x00000000 | Error state register 2 |
| CSI2HOST_MSK1 | 0x0028 | W | 0x00000000 | Masks for errors 1 |
| CSI2HOST_MSK2 | 0x002c | W | 0x00000000 | Masks for errors 2 |
| CSI2HOST_PHY_TEST_CTL0 | 0x0030 | W | 0x00000000 | DPHY Test interface control 0 |
| CSI2HOST_PHY_TEST_CTL1 | 0x0034 | W | 0x00000000 | DPHY Test interface control 1 |
| CSI2HOST_CONTROL | 0x0040 | W | 0x0c204000 | Control |

Notes: Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

14.4.2 Detail Register Description

CSI2HOST_VERSION

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RO | 0x0 | version Version of the DWC_mipi_csi2_host |

CSI2HOST_N_LANES

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:2 | RO | 0x0 | reserved |
| 1:0 | RW | 0x0 | n_lanes Number of active data lanes 00:1 data lane(lane 0) 01:2 data lanes(lane0 and 1) 10:3 data lanes(lane0,1,and 2) 11:4 data lanes(ALL) Can only be updated when the DPHY lane is in the Stop state |

CSI2HOST_dphy_shutdownz

Address: Operational Base + offset (0x0008)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | PHY_SHUTDOWNZ Shutdown input This line is used to place the complete macro in power down. All analog blocks are in power down mode and digital logic is cleared Active low |

CSI2HOST DPHY_RSTZ

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | dphy_rstz DPHY reset output Active low |

CSI2HOST CSI2 RESETN

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | csi2_resetn CSI2 controller reset output Active low |

CSI2HOST PHY STATE

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:12 | RO | 0x0 | reserved |
| 11 | RW | 0x0 | bypass_2ecc_tst Payload Bypass test mode for double ECC errors |
| 10 | RO | 0x0 | phy_stopstateclk Clock lane in Stop state |
| 9 | RO | 0x0 | phy_rxulpsclknot This signal indicates that the clock lane module has entered the Ultra Low Power state. Active low |
| 8 | RO | 0x0 | phy_rxclkactivehs Indicates that the clock lane is actively receiving a DDR clock |
| 7 | RO | 0x0 | phy_stopstatedata_3 Data lane 3 in Stop state |
| 6 | RO | 0x0 | phy_stopstatedata_2 Data lane 2 in Stop state |
| 5 | RO | 0x0 | phy_stopstatedata_1 Data lane 1 in Stop state |
| 4 | RO | 0x0 | phy_stopstatedata_0 Data lane 0 in Stop state |
| 3 | RO | 0x0 | phy_rxulpsesc_3 Lane module0 has entered the Ultra Low Power mode |
| 2 | RO | 0x0 | phy_rxulpsesc_2 Lane module2 has entered the Ultra Low Power mode |
| 1 | RO | 0x0 | phy_rxulpsesc_1 Lane module1 has entered the Ultra Low Power mode |
| 0 | RO | 0x0 | phy_rxulpsesc_0 Lane module0 has entered the Ultra Low Power mode |

CSI2HOST DATA IDS 1

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:30 | RW | 0x0 | di3_vc Data ID 3 virtual channel |
| 29:24 | RW | 0x00 | di3_dt Data ID 3 data type |
| 23:22 | RW | 0x0 | di2_vc Data ID 2 virtual channel |
| 21:16 | RW | 0x00 | di2_dt Data ID 2 data type |
| 15:14 | RW | 0x0 | di1_vc Data ID 1 virtual channel |
| 13:8 | RW | 0x00 | di1_dt Data ID 1 data type |
| 7:6 | RW | 0x0 | di0_vc Data ID 0 virtual channel |
| 5:0 | RW | 0x00 | di0_dt Data ID 0 data type |

CSI2HOST DATA IDS 2

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:30 | RW | 0x0 | di7_vc Data ID 7 virtual channel |
| 29:24 | RW | 0x00 | di7_dt Data ID 7 data type |
| 23:22 | RW | 0x0 | di6_vc Data ID 6 virtual channel |
| 21:16 | RW | 0x00 | di6_dt Data ID 6 data type |
| 15:14 | RW | 0x0 | di5_vc Data ID 5 virtual channel |
| 13:8 | RW | 0x00 | di5_dt Data ID 5 data type |
| 7:6 | RW | 0x0 | di4_vc Data ID 4 virtual channel |
| 5:0 | RW | 0x00 | di4_dt Data ID 4 data type |

CSI2HOST ERR1

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31 | RW | 0x0 | err_ph_crc_lane2 Packet crc error of lane2 when sw_cphy_en=1 |
| 30 | RW | 0x0 | err_ph_crc_lane1 Packet crc error of lane1 when sw_cphy_en=1 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 29 | RW | 0x0 | err_ph_crc_lane0 Packet crc error of lane0 when sw_cphy_en=1 |
| 28 | RO | 0x0 | err_ecc_double Header ECC contains 2 errors,unrecoverable |
| 27 | RO | 0x0 | vc3_err_crc Checksum error detected on virtual channel 3 |
| 26 | RO | 0x0 | vc2_err_crc Checksum error detected on virtual channel 2 |
| 25 | RO | 0x0 | vc1_err_crc Checksum error detected on virtual channel 1 |
| 24 | RO | 0x0 | vc0_err_crc Checksum error detected on virtual channel 0 |
| 23 | RO | 0x0 | err_l_seq_di3 Error in the sequence of lines for vc3 and dt3 |
| 22 | RO | 0x0 | err_l_seq_di2 Error in the sequence of lines for vc2 and dt2 |
| 21 | RO | 0x0 | err_l_seq_di1 Error in the sequence of lines for vc1 and dt1 |
| 20 | RO | 0x0 | err_l_seq_di0 Error in the sequence of lines for vc0 and dt0 |
| 19 | RO | 0x0 | err_l_bndry_match_di3 Error matching line start with line end for vc3 and dt3 |
| 18 | RO | 0x0 | err_l_bndry_match_di2 Error matching line start with line end for vc2 and dt2 |
| 17 | RO | 0x0 | err_l_bndry_match_di1 Error matching line start with line end for vc1 and dt1 |
| 16 | RO | 0x0 | err_l_bndry_match_di0 Error matching line start with line end for vc0 and dt0 |
| 15 | RO | 0x0 | err_frame_data_vc3 Last received frame,in virtual channel 3,had at least one CRC error |
| 14 | RO | 0x0 | err_frame_data_vc2 Last received frame,in virtual channel 2,had at least one CRC error |
| 13 | RO | 0x0 | err_frame_data_vc1 Last received frame,in virtual channel 1,had at least one CRC error |
| 12 | RO | 0x0 | err_frame_data_vc0 Last received frame,in virtual channel 0,had at least one CRC error |
| 11 | RO | 0x0 | err_f_seq_vc3 Incorrect frame sequence detected in virtual channel 3 |
| 10 | RO | 0x0 | err_f_seq_vc2 Incorrect frame sequence detected in virtual channel 2 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 9 | RO | 0x0 | err_f_seq_vc1 Incorrect frame sequence detected in virtual channel 1 |
| 8 | RO | 0x0 | err_f_seq_vc0 Incorrect frame sequence detected in virtual channel 0 |
| 7 | RO | 0x0 | err_f_bndry_match_vc3 Error matching frame start with frame end for virtual channel 3 |
| 6 | RO | 0x0 | err_f_bndry_match_vc2 Error matching frame start with frame end for virtual channel 2 |
| 5 | RO | 0x0 | err_f_bndry_match_vc1 Error matching frame start with frame end for virtual channel 1 |
| 4 | RO | 0x0 | err_f_bndry_match_vc0 Error matching frame start with frame end for virtual channel 0 |
| 3 | RO | 0x0 | phy_errsotsynchs_3 Start of transmission error on data lane 3 |
| 2 | RO | 0x0 | phy_errsotsynchs_2 Start of transmission error on data lane 2 |
| 1 | RO | 0x0 | phy_errsotsynchs_1 Start of transmission error on data lane 1 |
| 0 | RO | 0x0 | phy_errsotsynchs_0 Start of transmission error on data lane 0(no synchronization achieved) |

CSI2HOST_ERR2

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:28 | RO | 0x0 | reserved |
| 27 | RW | 0x0 | phy_errcodehs_3 lane 3 receiv error code |
| 26 | RW | 0x0 | phy_errcodehs_2 lane 2 receiv error code |
| 25 | RW | 0x0 | phy_errcodehs_1 lane 1 receiv error code |
| 24 | RW | 0x0 | phy_errcodehs_0 lane 0 receiv error code |
| 23 | RW | 0x0 | err_l_seq_di7 Error in the sequence of lines for vc7 and dt7 |
| 22 | RW | 0x0 | err_l_seq_di6 Error in the sequence of lines for vc6 and dt6 |
| 21 | RW | 0x0 | err_l_seq_di5 Error in the sequence of lines for vc5 and dt5 |
| 20 | RW | 0x0 | err_l_seq_di4 Error in the sequence of lines for vc4 and dt4 |
| 19 | RW | 0x0 | err_l_bndry_match_di7 Error matching line start with line end for vc7 and dt7 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 18 | RW | 0x0 | err_l_bndry_match_di6 Error matching line start with line end for vc6 and dt6 |
| 17 | RW | 0x0 | err_l_bndry_match_di5 Error matching line start with line end for vc5 and dt5 |
| 16 | RW | 0x0 | err_l_bndry_match_di4 Error matching line start with line end for vc4 and dt4 |
| 15 | RW | 0x0 | err_id_vc3 Unrecognized or unimplemented data type detected in virtual channel 3 |
| 14 | RW | 0x0 | err_id_vc2 Unrecognized or unimplemented data type detected in virtual channel 2 |
| 13 | RW | 0x0 | err_id_vc1 Unrecognized or unimplemented data type detected in virtual channel 1 |
| 12 | RW | 0x0 | err_id_vc0 Unrecognized or unimplemented data type detected in virtual channel 0 |
| 11 | RW | 0x0 | vc3_err_ecc_corrected Header error detected and corrected on virtual channel 3 |
| 10 | RW | 0x0 | vc2_err_ecc_corrected Header error detected and corrected on virtual channel 2 |
| 9 | RW | 0x0 | vc1_err_ecc_corrected Header error detected and corrected on virtual channel 1 |
| 8 | RW | 0x0 | vc0_err_ecc_corrected Header error detected and corrected on virtual channel 0 |
| 7 | RW | 0x0 | phy_errsoths_3 Start of transmission error on data lane 3(synchronization can still be achieved) |
| 6 | RW | 0x0 | phy_errsoths_2 Start of transmission error on data lane 2(synchronization can still be achieved) |
| 5 | RW | 0x0 | phy_errsoths_1 Start of transmission error on data lane 1(synchronization can still be achieved) |
| 4 | RW | 0x0 | phy_errsoths_0 Start of transmission error on data lane 0(synchronization can still be achieved) |
| 3 | RW | 0x0 | phy_erresc_3 Escape entry error(ULPM) on data lane 3 |
| 2 | RW | 0x0 | phy_erresc_2 Escape entry error(ULPM) on data lane 2 |
| 1 | RW | 0x0 | phy_erresc_1 Escape entry error(ULPM) on data lane 1 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 0 | RW | 0x0 | phy_erresc_0 Escape entry error(ULPM) on data lane 0 |

CSI2HOST_MSK1

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31 | RW | 0x0 | mask_err_ph_crc_lane2 Mask for err_ph_crc_lane2 |
| 30 | RW | 0x0 | mask_err_ph_crc_lane1 Mask for err_ph_crc_lane1 |
| 29 | RW | 0x0 | mask_err_ph_crc_lane0 Mask for err_ph_crc_lane0 |
| 28 | RW | 0x0 | mask_err_ecc_double Mask for err_ecc_double |
| 27 | RW | 0x0 | mask_vc3_err_crc Mask for vc3_err_crc |
| 26 | RW | 0x0 | mask_vc2_err_crc Mask for vc2_err_crc |
| 25 | RW | 0x0 | mask_vc1_err_crc Mask for vc1_err_crc |
| 24 | RW | 0x0 | mask_vc0_err_crc Mask for vc0_err_crc |
| 23 | RW | 0x0 | mask_err_l_seq_di3 Mask for err_l_seq_di3 |
| 22 | RW | 0x0 | mask_err_l_seq_di2 Mask for err_l_seq_di2 |
| 21 | RW | 0x0 | mask_err_l_seq_di1 Mask for err_l_seq_di1 |
| 20 | RW | 0x0 | mask_err_l_seq_di0 Mask for err_l_seq_di0 |
| 19 | RW | 0x0 | mask_err_l_bndry_match_di3 Mask for err_l_bndry_match_di3 |
| 18 | RW | 0x0 | mask_err_l_bndry_match_di2 Mask for err_l_bndry_match_di2 |
| 17 | RW | 0x0 | mask_err_l_bndry_match_di1 Mask for err_l_bndry_match_di1 |
| 16 | RW | 0x0 | mask_err_l_bndry_match_di0 Mask for err_l_bndry_match_di0 |
| 15 | RW | 0x0 | mask_err_frame_data_vc3 Mask for err_frame_data_vc3 |
| 14 | RW | 0x0 | mask_err_frame_data_vc2 Mask for err_frame_data_vc2 |
| 13 | RW | 0x0 | mask_err_frame_data_vc1 Mask for err_frame_data_vc1 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 12 | RW | 0x0 | mask_err_frame_data_vc0 Mask for err_frame_data_vc0 |
| 11 | RW | 0x0 | mask_err_f_seq_vc3 Mask for err_f_seq_vc3 |
| 10 | RW | 0x0 | mask_err_f_seq_vc2 Mask for err_f_seq_vc2 |
| 9 | RW | 0x0 | mask_err_f_seq_vc1 Mask for err_f_seq_vc1 |
| 8 | RW | 0x0 | mask_err_f_seq_vc0 Mask for err_f_seq_vc0 |
| 7 | RW | 0x0 | mask_err_f_bndry_match_vc3 Mask for err_f_bndry_match_vc3 |
| 6 | RW | 0x0 | mask_err_f_bndry_match_vc2 Mask for err_f_bndry_match_vc2 |
| 5 | RW | 0x0 | mask_err_f_bndry_match_vc1 Mask for err_f_bndry_match_vc1 |
| 4 | RW | 0x0 | mask_err_f_bndry_match_vc0 Mask for err_f_bndry_match_vc0 |
| 3 | RW | 0x0 | mask_phy_errsotsynchs_3 Mask for phy_errsotsynchs_3 |
| 2 | RW | 0x0 | mask_phy_errsotsynchs_2 Mask for phy_errsotsynchs_2 |
| 1 | RW | 0x0 | mask_phy_errsotsynchs_1 Mask for phy_errsotsynchs_1 |
| 0 | RW | 0x0 | mask_phy_errsotsynchs_0 Mask for phy_errsotsynchs_0 |

CSI2HOST MSK2

Address: Operational Base + offset (0x002c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:24 | RO | 0x0 | reserved |
| 23 | RW | 0x0 | mask_err_l_seq_di7 Mask for err_l_seq_di7 |
| 22 | RW | 0x0 | mask_err_l_seq_di6 Mask for err_l_seq_di6 |
| 21 | RW | 0x0 | mask_err_l_seq_di5 Mask for err_l_seq_di5 |
| 20 | RW | 0x0 | mask_err_l_seq_di4 Mask for err_l_seq_di4 |
| 19 | RW | 0x0 | mask_err_l_bndry_match_di7 Mask for err_l_bndry_match_di7 |
| 18 | RW | 0x0 | mask_err_l_bndry_match_di6 Mask for err_l_bndry_match_di6 |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 17 | RW | 0x0 | mask_err_l_bndry_match_di5 Mask for err_l_bndry_match_di5 |
| 16 | RW | 0x0 | mask_err_l_bndry_match_di4 Mask for err_l_bndry_match_di4 |
| 15 | RW | 0x0 | mask_err_id_vc3 Mask for err_id_vc3 |
| 14 | RW | 0x0 | mask_err_id_vc2 Mask for err_id_vc2 |
| 13 | RW | 0x0 | mask_err_id_vc1 Mask for err_id_vc1 |
| 12 | RW | 0x0 | mask_err_id_vc0 Mask for err_id_vc0 |
| 11 | RW | 0x0 | mask_vc3_err_ecc_corrected Mask for vc3_err_ecc_corrected |
| 10 | RW | 0x0 | mask_vc2_err_ecc_corrected Mask for vc2_err_ecc_corrected |
| 9 | RW | 0x0 | mask_vc1_err_ecc_corrected Mask for vc1_err_ecc_corrected |
| 8 | RW | 0x0 | mask_vc0_err_ecc_corrected Mask for vc0_err_ecc_corrected |
| 7 | RW | 0x0 | mask_phy_errsoths_3 Mask for phy_errsoths_3 |
| 6 | RW | 0x0 | mask_phy_errsoths_2 Mask for phy_errsoths_2 |
| 5 | RW | 0x0 | mask_phy_errsoths_ Mask for phy_errsoths_ |
| 4 | RW | 0x0 | mask_phy_errsoths_0 Mask for phy_errsoths_0 |
| 3 | RW | 0x0 | mask_phy_erresc_3 Mask for phy_erresc_3 |
| 2 | RW | 0x0 | mask_phy_erresc_2 Mask for phy_erresc_2 |
| 1 | RW | 0x0 | mask_phy_erresc_1 Mask for phy_erresc_1 |
| 0 | RW | 0x0 | mask_phy_erresc_0 Mask for phy_erresc_0 |

CSI2HOST PHY TEST CTRL0

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:2 | RO | 0x0 | reserved |

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 1 | RW | 0x0 | phy_testclk DPHY test interface strobe signal It is used to clock TESTDIN bus into the DPHY.In conjunction with TESTEN signal controls the operation selection |
| 0 | RW | 0x0 | phy_testclr DPHY test interface clear It is used when active performs vendor specific interface initialization Active High |

CSI2HOST PHY TEST CTRL1

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:17 | RO | 0x0 | reserved |
| 16 | RW | 0x0 | phy_testen DPHY test interface operation selector: 1: Configures address write operation on the falling edge of TESTCLK 0: Configures a data write operation on the rising edge of TESTCLK |
| 15:8 | RW | 0x00 | phy_testdout DPHY output 8-bit data bus for read-back and internal probing functionalities |
| 7:0 | RW | 0x00 | phy_testdin DPHY test interface input 8-bit bus for internal register programming and test functionalities access |

CSI2HOST CONTROL

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:26 | RW | 0x03 | sw_datatype_le The datatype of line end |
| 25:20 | RW | 0x02 | sw_datatype_ls The datatype of line start |
| 19:14 | RW | 0x01 | sw_datatype_fe The datatype of frame end |
| 13:8 | RW | 0x00 | sw_datatype_fs The datatype of frame start |
| 7:5 | RO | 0x0 | reserved |
| 4 | RW | 0x0 | sw_dsi_en 1'b0: For csi2 1'b1: For dsi |
| 3:1 | RO | 0x0 | reserved |
| 0 | RW | 0x0 | sw_cphy_en 1'b0: For dphy 1'b1: For cphy |

14.5 Application Notes

The most important step is configuring the right CSI2HOST_N_LANES before pulling up csi_resetn. If the host is used to receive DSI data, the sw_dsi_en must be enabled, and the sw_datatype_fs, sw_datatype_fe, sw_datatype_ls, sw_datatype_le must be configured correctly. When the sw_dsi_en is enable, those debug or error detection registers are useless.

NOTE: The CSI2HOST_DPHY_SHUTDOWNZ, CSI2HOST_DPHY_RSTZ, CSI2HOST_PHY_TEST_CTRL0, CSI2HOST_PHY_TEST_CTRL1 are useless

Chapter 15 MIPI DPHY TX

15.1 Overview

The MIPI DPHY TX integrates a MIPI® V1.2 compatible PHY that supports up to 2.0Gbps high speed data transmitter, plus a MIPI® low-power low speed transceiver that supports data transfer in the bi-directional mode. The IP supports the full specifications described in V1.2 of the DPHY spec. The DPHY is built in with a standard digital interface to talk to any third party Host controller. The architecture supports connection of multiple data lanes in parallel – up to 4 data lanes can be connected to increase the total through-put, customizable to user determined configurations. The I/O and ESD are also built-in as one in a rectangular footprint for any configuration (one lane to 4 lanes or more). It is optimized for high speed applications with robust timing and small silicon area. The DPHY supports the electrical portion of MIPI DPHY V1.0 standard, covering all transmission modes (ULP/LP/HS).The MIPI DPHY cost-effectively adds MIPI DPHY V1.2 capability to any SOC used in communication and consumer electronics field.

The MIPI DPHY supports the following features:

- Mixed-signal DPHY mixed-signal hard-macro- HS/LS Transmitter and LS/HS Receiver solution
- Designed to MIPI® v1.2 Specifications
- Integrated PHY Protocol Interface (PPI) supports interface to CSI, DSI and UniPro™
- MIPI® protocols
- 2.0Gbps maximum data transfer rate per lane
- Expandable to support 4 data lanes, providing up to 8.0Gbps transfer rate
- MIPI-HS, MIPI-LP modes supported
- Skew-calibration supported
- 10Mbps per lane in low-power mode
- Unidirectional and bi-directional modes supported
- Automatic termination control for HS and LP modes
- Low-Power dissipation: less than 22mA/Lane in HS TX/RX mode
- Tx/Rx Buffers with tunable On-Die-Termination and advanced equalization.
- Embedded ESD, boundary scan support logic.

15.2 Block Diagram

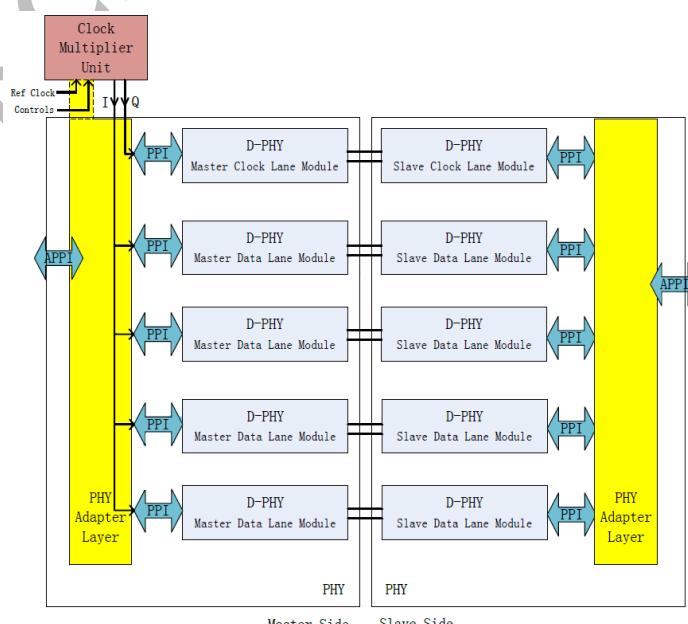


Fig. 15-1 MIPI DPHY TX detailed block diagram

MIPI DPHY configuration contains one Clock Lane Module and four Data Lane Modules. Each of these PHY Lane Modules communicates via two Lines to a complementary part at the other side of the Lane Interconnect. Figure 1 shows the DPHY architecture.

15.3 Function Description

15.3.1 System Connection

The MIPI DPHY supports DSI/CSI mode as shown in following figure:

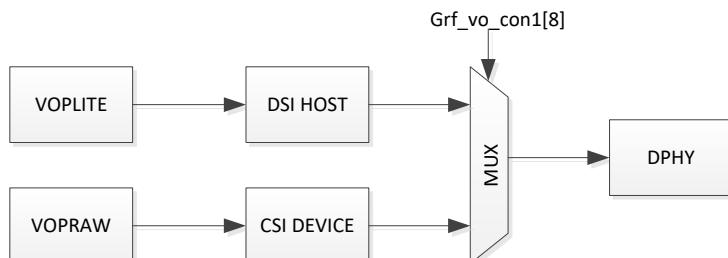


Fig. 15-2 MIPI DPHY TX VOPLITE and VOPRAW connection

15.3.2 MIPI MODE

In MIPI mode, there are some functions of DPHY controlled by grf registers. These register bit and function is show in following table:

Table 15-1 Function of grf bits in MIPI mode

| register bit | function |
|-----------------------------------|---|
| grf_con_dsi_phy_lane0_frctxstpm | Force Lane Module Into Transmit mode / Generate Stop state. This signal allows the protocol to force a Lane Module into transmit mode and Stop state during initialization. |
| grf_con_dsi_phy_lane1_frctxstpm | Force Lane Module Into Transmit mode / Generate Stop state. This signal allows the protocol to force a Lane Module into transmit mode and Stop state during initialization. |
| grf_con_dsi_phy_lane2_frctxstpm | Force Lane Module Into Transmit mode / Generate Stop state. This signal allows the protocol to force a Lane Module into transmit mode and Stop state during initialization. |
| grf_con_dsi_phy_lane3_frctxstpm | Force Lane Module Into Transmit mode / Generate Stop state. This signal allows the protocol to force a Lane Module into transmit mode and Stop state during initialization. |
| grf_con_dsi_phy_lane0_turndisable | Disable Turn-around. This signal is used to prevent a (bi-directional) Lane from going into transmit mode – even if it observes a turn-around request on the Lane interconnect. |
| grf_con_dsi_phy_forcerxmode | Force Lane Module Into Receive mode / Wait for Stop state. This signal allows the protocol to initialize a Lane Module, or force a bi-directional Lane Module, into receive mode. |

15.3.3 Program PLL in MIPI mode

Frequency Calculating Formula

The PLL output frequency can be calculated using a simple formula:

$$\text{PLL_Output_Frequency} = (\text{FREF}/\text{PREDIV} * \text{FBDIV})/\text{POSTDIV}$$

PLL_Output_Frequency: It is equal to DDR_Clock_Frequency * 2

FREF: 24 Mhz

PREDIV: PLL input reference clock divider which can be configured by the register of prediv

FBDIV: Integer value programmed into feedback divider which can be configured by the register of fbdv

POSTDIV:PLL post-divider 2 can be configured by the registers of reg_postdiv . If

reg_postdiv is set to 1 , the POSTDIV value is 2 , otherwise , the value is 1 .

Additional Programming Considerations

1. The divided reference frequency (FREF/PREDIV) should be less than 40MHz.

2. The all possible settings of feedback divider are 12, 13, 14, 16~511.

15.4 Register Description

DSI PHY registers are accessed by APB BUS. This section describes the control/status registers of the design.

15.4.1 Register Summary

| Name | Offset | Size | Reset Value | Description |
|------------------------------|--------|------|-------------|---------------------|
| MIPI DSIPHY CTRL LANE ENABLE | 0x0000 | W | 0x00000001 | Lane enable |
| MIPI DSIPHY CTRL PWR CTL | 0x0004 | W | 0x00000003 | Power control |
| MIPI DSIPHY CTRL PRED IV | 0x000c | W | 0x00000003 | PLL PREDIV |
| MIPI DSIPHY CTRL FBDIV | 0x0010 | W | 0x0000007d | PLL FBDIV |
| MIPI DSIPHY CTRL DIG RST | 0x0080 | W | 0x000000ff | Digital logic reset |
| MIPI DSIPHY CTRL SIG INV | 0x0084 | W | 0x00000002 | Signal revert |

Notes: Size : **B** - Byte (8 bits) access, **HW** - Half WORD (16 bits) access, **W** -WORD (32 bits) access.

Notes: Size : **B** - Byte (8 bits) access, **HW** - Half WORD (16 bits) access, **W** -WORD (32 bits) access

Following table shows registers for 5 lanes. They have same registers but use different address bases. So their final address is lane_base_address + offset.

| lane | lane_base_address |
|------------|-------------------|
| Clock_lane | 0x100 |
| Data0_lane | 0x180 |
| Data1_lane | 0x200 |
| Data2_lane | 0x280 |
| Data3_lane | 0x300 |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------|--------|------|-------------|-------------|
| MIPI DSIPHY LANEX TLP X | 0x0014 | W | 0x000000c5 | TLPX |
| MIPI DSIPHY LANEX S HSTXTHSPRPR | 0x0018 | W | 0x00000024 | HSTXTHSPRPR |

| Name | Offset | Size | Reset Value | Description |
|---------------------------------------|--------|------|-------------|--------------------|
| MIPI DSIPHY LANEX_S_HSTXTHSZERO | 0x001c | W | 0x00000000 | HSTXTHSZERO |
| MIPI DSIPHY LANEX_S_HSTX | 0x0020 | W | 0x00000027 | HSTX |
| MIPI DSIPHY LANEX_THS_EXIT | 0x0024 | W | 0x0000000a | THS_EXIT |
| MIPI DSIPHY LANEX_TCLK_POST | 0x0028 | W | 0x0000000f | TCLK_POST |
| MIPI DSIPHY LANEX_LPD_T_TX_PPI_SYN_EN | 0x0030 | W | 0x00000000 | LPDT_TX_PPI_SYN_EN |
| MIPI DSIPHY LANEX_TWAKUP | 0x0034 | W | 0x00000000 | TWAKUP |
| MIPI DSIPHY LANEX_TCLK_PRE | 0x0038 | W | 0x00000006 | TCLK_PRE |
| MIPI DSIPHY LANEX_TTA_GO | 0x0040 | W | 0x00000004 | TTA_GO |
| MIPI DSIPHY LANEX_TTA_SURE | 0x0044 | W | 0x00000001 | TTA_SURE |
| MIPI DSIPHY LANEX_TTA_WAIT | 0x0048 | W | 0x00000032 | TTA_WAIT |

Notes: Size : **B** - Byte (8 bits) access, **HW** - Half WORD (16 bits) access, **W** - WORD (32 bits) access

15.4.2 MIPI DSIPHY CTRL Detail Register Description

MIPI DSIPHY CTRL LANE ENABLE

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:8 | RO | 0x0 | reserved |
| 7 | RW | 0x0 | Bandgap power down enable 1'b0:Power on 1'b1:Power down |
| 6 | RW | 0x0 | lane_en_ck Lane clk enable |
| 5 | RW | 0x0 | lane_en_3 Lane 3 enable |
| 4 | RW | 0x0 | lane_en_2 Lane 2 enable |
| 3 | RW | 0x0 | Lane_en_1 Lane 1 enable |
| 2 | RW | 0x0 | lane_en_0 Lane 0 enable |
| 1:0 | RO | 0x0 | Power work 2'b0x:Power work enable 2'b10:Power work disable 2'b11:Reserved |

MIPI DSIPHY CTRL PWRCTL

Address: Operational Base + offset (0x0004)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------|
| 31:3 | RO | 0x0 | Reserved |
| 2 | RW | 0x0 | syncrst Analog reset |
| 1 | RW | 0x1 | ldo_pd LDO powerdown |
| 0 | RW | 0x1 | pll_pd PLL power down |

MIPI DSIPHY CTRL PREDIV

Address: Operational Base + offset (0x000c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:6 | RO | 0x0 | Reserved |
| 5 | RW | 0x0 | fbdv_hi |
| 4:0 | RW | 0x03 | pre_div |

MIPI DSIPHY CTRL FBDIV

Address: Operational Base + offset (0x0010)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0x7d | fbdv_lo |

MIPI DSIPHY CTRL DIG_RST

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------------------|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x1 | dig_rst Digital logic reset |

MIPI DSIPHY CTRL SIG INV

Address: Operational Base + offset (0x0084)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:2 | RO | 0x0 | Reserved |
| 1 | RW | 0x0 | Pin_txclkesc_0_inv_en Inverting pin_txclkesc_0 enable |
| 0 | RW | 0x0 | PIN_TXBUTECLKHS_INV_EN pin_txbyteclkhs_inv_en Inverting pin_txbyteclkhs enable |

15.4.3 MIPI DSIPHY LANEx Detail Register Description**MIPI DSIPHY LANEX TLPX**

Address: Operational Base + offset (0x0014)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x00 | tlpx The value of counter for HS tlpx Time (>=tlpx) = tpin_txbyteclkhs * (2+value) |

MIPI DSIPHY LANEX S HSTXTHSPRPR

Address: Operational Base + offset (0x0018)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | Reserved |
| 6:0 | RW | 0x24 | s_hstxthsprpr The value of counter for HS Ths-prepare For clock lane, ths-prepare(38ns~95ns) For data lane, ths-prepare(40ns+4UI~85ns+6UI) For clock lane, s_hstxthsprpr[6:0] = 7'b00000011 For data lane, s_hstxthsprpr[6:0] = 7'b00000011 Frequency(1/UI) value(HEX) 80-110MHz 7f 110-150 MHz 7f 150-200 MHz 7f 200-250 MHz 7f 250-300 MHz 7f 300-400 MHz 7e 400-500 MHz 70 500-600 MHz 60 600-700 MHz 40 700-800 MHz 02 800-1000 MHz 08 1000-1200Mhz 03 1200-1400Mhz 03 1400-1600Mhz 42 1600-1800Mhz 47 1800-2000Mhz 64 |

MIPI DSIPHY LANEX S HSTXTHSZERO

Address: Operational Base + offset (0x001c)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:6 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description |
|-----|------|-------------|--|
| 5:0 | RW | 0x00 | <p>s_hstxthszero For clock lane, Ths-prepare+Ths-zero (>=300ns) For data lane, Ths-prepare+Ths-zero (>= 145 ns + 10*UI) = Tpin_txbyteclkhs*(5+value) For clock lane, s_hstxthszero[5:0] = 6'b100000 Frequency(1/UI) Value(HEX)</p> <p>80 -110 MHz 16 110-150 MHz 16 150-200 MHz 17 200-250 MHz 17 250-300 MHz 18 300-400 MHz 19 400-500 MHz 1B 500-600 MHz 1D 600-700 MHz 1E 700-800 MHz 1F 800-1000 MHz 20 1000-1200Mhz 32 1200-1400Mhz 32 1400-1600Mhz 36 1600-1800Mhz 7a 1800-2000Mhz 7a</p> <p>For data lane,s_hstxthszero[5:0] = 6'b001001 Frequency(1/UI) Value(HEX)</p> <p>80 -110 MHz 2 110-150 MHz 3 150-200 MHz 4 200-250 MHz 5 250-300 MHz 6 300-400 MHz 7 400-500 MHz 7 500-600 MHz 8 600-700 MHz 8 700-800 MHz 9 800-1000 MHz 9 1000-1200Mhz 14 1200-1400Mhz 14 1400-1600Mhz 0e 1600-1800Mhz 0e 1800-2000Mhz 0e</p> |

MIPI DSIPHY LANEX S HSTX

Address: Operational Base + offset (0x0020)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | Reserved |
| 6:0 | RW | 0x27 | <p>s_hstxthstrail The value of counter for HS Ths-trail For clock lane, Ths-trail (\geq60ns) For data lane, Ths-trail (\geqmax(8UI, 60ns+4UI)) For clock lane, s_hstxthstrail[6:0] = 7'b0000111 For data lane, s_hstxthstrail[6:0] = 7'b0000111 Frequency(1/UI) Value(HEX) 80 -110 MHz 02 110-150 MHz 02 150-200 MHz 02 200-250 MHz 04 250-300 MHz 04 300-400 MHz 04 400-500 MHz 08 500-600 MHz 10 600-700 MHz 30 700-800 MHz 30 800-1000 MHz 30 1000-1200MHz 0f 1200-1400MHz 0f 1400-1600MHz 0f 1600-1800MHz 0f 1800-2000MHz 0b</p> |

MIPI DSIPHY LANEX THS EXIT

Address: Operational Base + offset (0x0024)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:5 | RO | 0x0 | Reserved |
| 4:0 | RW | 0x0a | <p>ths_exit The value of counter for HS ths-exit Ths-exit = Tpin_txbyteclkhs*value</p> |

MIPI DSIPHY LANEX TCLK POST

Address: Operational Base + offset (0x0028)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:4 | RO | 0x0 | Reserved |
| 3:0 | RW | 0xf | <p>tclk_post The value of counter for HS tclk-post Tclk-post = Tpin_txbyteclkhs*value</p> |

MIPI DSIPHY LANEX LPDT TX PPI SYN EN

Address: Operational Base + offset (0x0030)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:3 | RO | 0x0 | Reserved |
| 2 | RW | 0x0 | LPDT_TX_PPI_SYN_EN lpdt_tx_ppi_syn_en LPDT TX PPI signals internal synchronization enable. 1'b0: Disable 1'b1: Enable |
| 1:0 | RW | 0x0 | twakeup_h The value[9:8] of counter for HS twakeup |

MIPI DSIPHY LANEX TWAKUP

Address: Operational Base + offset (0x0034)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7:0 | RW | 0x00 | twakeup The value[7:0] of counter for HS twakeup Twakup for ulpm, Twakup = Tpin_sys_clk*value[9:0] |

MIPI DSIPHY LANEX TCLK PRE

Address: Operational Base + offset (0x0038)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:4 | RO | 0x0 | Reserved |
| 3:0 | RW | 0x6 | tclk_pre |

MIPI DSIPHY LANEX TTA GO

Address: Operational Base + offset (0x0040)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x04 | tta_go The value of counter for HS tta-go Tta-go for turnaround Tta-go = Ttxclkesc*value |

MIPI DSIPHY LANEX TTA SURE

Address: Operational Base + offset (0x0044)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x01 | tta_sure The value of counter for HS tta-sure Tta-sure for turnaround |

MIPI DSIPHY LANEX TTA WAIT

Address: Operational Base + offset (0x0048)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x00 | tta_wait The value of counter for HS tta-wait Tta-wait for turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value |

15.5 Application Notes

15.5.1 COMMON CONFIGURATION (DEFAULT IN MIPI MODE)

Step1: Enable reference clock.

Step2: Wait a period after reference clock have been enabled.

Step3: Send 0xe4 to register MIPI_DSI_DPHY_CTRL_PWRCTL. Enable PLL and LDO.

Step4: Send 0x7d to register MIPI_DSI_DPHY_CTRL_LANE_ENABLE. Enable all lanes on analog part.

Step5: Send 0xe0 to register MIPI_DSI_DPHY_CTRL_PWRCTL. Reset analog.

Step6: Wait a period after analog has been reset.

Step7: Send 0x1e to register MIPI_DSI_DPHY_CTRL_DIG_RST. Reset digital.

Step8: Send 0x1f to register MIPI_DSI_DPHY_CTRL_DIG_RST. Reset digital.

Step9: Wait a period until pll locked. Run in MIPI mode.

15.5.2 LOW POWER MODE (FOR DSI ONLY)

Low Power Mode is a special feature for DPHY. You can control this function by using proper registers from the DPHY with few operations. The following is a step by step instruction for low power mode in and out.

● Low Power in Steps:

Step1: Send 0x01 to register MIPI_DSI_DPHY_CTRL_LANE_ENABLE. Disable all lanes on analog part.

Step2: Send 0xe3 to register MIPI_DSI_DPHY_CTRL_PWRCTL. Disable PLL and LDO.

Step3: Wait a period before reference clock have been disabled.

Step4: Disable reference clock.

● Low Power out Steps:

Step1: Enable reference clock.

Step2: Wait a period after reference clock have been enabled.

Step3: Send 0xe4 to register MIPI_DSI_DPHY_CTRL_PWRCTL. Enable PLL and LDO.

Step4: Send 0x7d to register MIPI_DSI_DPHY_CTRL_LANE_ENABLE. Enable all lanes on analog part.

Step5: Send 0xe0 to register MIPI_DSI_DPHY_CTRL_PWRCTL. Reset analog.

Step6: Wait a period after analog has been reset.

Step7: Send 0x1e to register MIPI_DSI_DPHY_CTRL_DIG_RST. Reset digital.

Step8: Send 0x1f to register MIPI_DSI_DPHY_CTRL_DIG_RST. Reset digital.

Step9: Wait a period before normal transmission.

Chapter 16 MIPI DPHY RX

16.1 Overview

The features of MIPI DPHY RX are as follow:

- Analog mixed-signal hard-macro LP/HS receiver solution
- Designed to MIPI v1.2 Specifications
- Integrated PHY Protocol Interface(PPI) interfaces to DSI/CSI and UniPro MIPI protocols
- 2 Gbps maximum data transfer rate per lane
- Expandable to support 4 data lanes, providing up to 8 Gbps transfer rate
- HS-RX,LP-RX,LP-TX and Calibration supported
- 10Mbps per lane in low-power mode
- Unidirectional and bi-directional modes supported
- Automatic termination control for HS and LP modes
- LOW-Power dissipation: less than 1.5mA/Lane in DPHY HS RX mode
- Buffers with tunable On-Die-Termination
- Includes embedded ESD, boundary scan and BIST

16.2 Block Diagram

MIPI DPHY configuration contains one Clock Lane Module and four Data Lane Modules. Each of these PHY Lane Modules communicates via two Lines to a complementary part at the other side of the Lane Interconnect. Figure below shows a Universal Lane Module Diagram with a global overview of internal functionality of the CIL function. This Universal Module can be used for all Lane types. The requirements for the 'Control and Interface Logic'(CIL) function depend on the Lane type and Lane side.

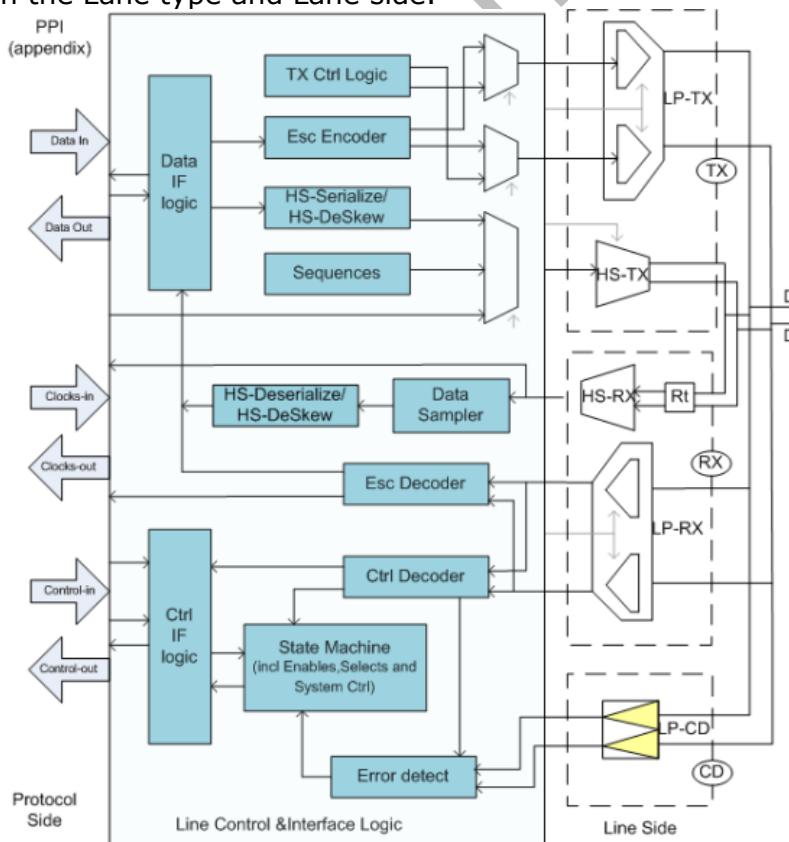


Fig. 16-1 MIPI DPHY RX Block Diagram

16.3 Register Description

16.3.1 Registers Summary

| Name | Offset | Size | Reset Value | Description |
|----------------------------|--------|------|-------------|---|
| DPHY RX LANE EN | 0x0000 | W | 0x00000000 | Clk/data lane enable |
| DPHY RX DIG RSTN | 0x0080 | W | 0x00000000 | Dig reset |
| DPHY RX LANE CK MOD E | 0x0128 | W | 0x00000000 | Clock lane mode |
| DPHY RX LANE CK MSB | 0x0138 | W | 0x00000000 | MSB enable |
| DPHY RX LANE CK TTAG O | 0x0140 | W | 0x00000004 | The value of counter for Tta-go of turnaround |
| DPHY RX LANE CK TTAS URE | 0x0144 | W | 0x00000001 | The value of counter for Tta-sure of turnaround |
| DPHY RX LANE CK TTA WAIT | 0x0148 | W | 0x00000032 | The value of counter for Tta-wait of turnaround |
| DPHY RX LANE CK THS SETTLE | 0x0160 | W | 0x0000001b | The count time of the THS-SETTLE by protocol |
| DPHY RX LANE CK CAL EN | 0x0168 | W | 0x00000000 | Calibration reception enable |
| DPHY RX LANE 0 MSB | 0x01b8 | W | 0x00000000 | MSB enable |
| DPHY RX LANE 0 TTAG O | 0x01c0 | W | 0x00000004 | The value of counter for Tta-go of turnaround |
| DPHY RX LANE 0 TTASU RE | 0x01c4 | W | 0x00000001 | The value of counter for Tta-sure of turnaround |
| DPHY RX LANE 0 TTAW AIT | 0x01c8 | W | 0x00000032 | The value of counter for Tta-wait of turnaround |
| DPHY RX LANE 0 THSSE TTLE | 0x01e0 | W | 0x0000001b | The count time of the THS-SETTLE by protocol |
| DPHY RX LANE 0 CAL E N | 0x01e8 | W | 0x00000000 | Calibration reception enable |
| DPHY RX LANE 1 MSB | 0x0238 | W | 0x00000000 | MSB enable |
| DPHY RX LANE 1 TTAG Q | 0x0240 | W | 0x00000004 | The value of counter for Tta-go of turnaround |
| DPHY RX LANE 1 TTASU RE | 0x0244 | W | 0x00000001 | The value of counter for Tta-sure of turnaround |
| DPHY RX LANE 1 TTAW AIT | 0x0248 | W | 0x00000032 | The value of counter for Tta-wait of turnaround |
| DPHY RX LANE 1 THSSE TTLE | 0x0260 | W | 0x0000001b | The count time of the THS-SETTLE by protocol |
| DPHY RX LANE 1 CAL E N | 0x0268 | W | 0x00000000 | Calibration reception enable |
| DPHY RX LANE 2 MSB | 0x02b8 | W | 0x00000000 | MSB enable |
| DPHY RX LANE 2 TTAG Q | 0x02c0 | W | 0x00000004 | The value of counter for Tta-go of turnaround |

| Name | Offset | Size | Reset Value | Description |
|--------------------------|--------|------|-------------|---|
| DPHY RX LANE 2 TTASURE | 0x02c4 | W | 0x00000001 | The value of counter for Tta-sure of turnaround |
| DPHY RX LANE 2 TTAWAIT | 0x02c8 | W | 0x00000032 | The value of counter for Tta-wait of turnaround |
| DPHY RX LANE 2 THSSETTLE | 0x02e0 | W | 0x0000001b | The count time of the THS-SETTLE by protocol |
| DPHY RX LANE 2 CAL EN | 0x02e8 | W | 0x00000000 | Calibration reception enable |
| DPHY RX LANE 3 MSB | 0x0338 | W | 0x00000000 | MSB enable |
| DPHY RX LANE 3 TTAGO | 0x0340 | W | 0x00000004 | The value of counter for Tta-go of turnaround |
| DPHY RX LANE 3 TTASURE | 0x0344 | W | 0x00000001 | The value of counter for Tta-sure of turnaround |
| DPHY RX LANE 3 TTAWAIT | 0x0348 | W | 0x00000032 | The value of counter for Tta-wait of turnaround |
| DPHY RX LANE 3 THSSETTLE | 0x0360 | W | 0x0000001b | The count time of the THS-SETTLE by protocol |
| DPHY RX LANE 3 CAL EN | 0x0368 | W | 0x00000000 | Calibration reception enable |

Notes:Size: **B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access

16.3.2 Detail Register Description

DPHY RX LANE EN

Address: Operational Base + offset (0x0000)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | lane_en_ck Enable DPHY clock lane:active high |
| 5 | RW | 0x0 | lane_en_3 Enable DPHY lane3:active high |
| 4 | RW | 0x0 | lane_en_2 Enable DPHY lane2:active high |
| 3 | RW | 0x0 | lane_en_1 Enable DPHY lane1:active high |
| 2 | RW | 0x0 | lane_en_0 Enable DPHY lane0:active high |
| 1:0 | RO | 0x0 | Reserved |

DPHY RX DIG RSTN

Address: Operational Base + offset (0x0080)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|-------------------------------------|
| 31:1 | RO | 0x0 | Reserved |
| 0 | RW | 0x0 | reg_dig_rstn 1:Normal 0:Reset |

DPHY RX LANE CK MODE

Address: Operational Base + offset (0x0128)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:4 | RW | 0x0 | lane_ck_mode 2'b11:Enable continuous clock mode 2'b00:Disable continuous clock mode |
| 3:0 | RO | 0x0 | Reserved |

DPHY RX LANE CK MSB

Address: Operational Base + offset (0x0138)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | lane_ck_msb MSB enable for pin_rxdatahs_* 1:Enable 0:Disable |
| 5:0 | RO | 0x0 | Reserved |

DPHY RX LANE CK TTAGO

Address: Operational Base + offset (0x0140)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x04 | lane_ck_ttago The value of counter for Tta-go of turnaround Tta-go = Ttxclkesc*value |

DPHY RX LANE CK TTASURE

Address: Operational Base + offset (0x0144)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x01 | lane_ck_ttasure The value of counter for Tta-sure of turnaround Tta-sure = Ttxclkesc*value |

DPHY RX LANE CK TTAWAIT

Address: Operational Base + offset (0x0148)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x32 | <p>lane_ck_ttawait The value of counter for Tta-wait of turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value</p> |

DPHY RX LANE CK THSSETTLE

Address: Operational Base + offset (0x0160)

| Bit | Attr | Reset Value | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|-------------|--------------------|---|-----------------|------------|-----------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|-------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|
| 31:6 | RO | 0x0 | Reserved | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5:0 | RW | 0x1b | <p>lane_ck_thssettle The count time of the THS-SETTLE by protocol. after the count done, DPHY will begin to receive the high speed data.</p> <table> <thead> <tr> <th>Frequency(1/UI)</th> <th>Value(HEX)</th> </tr> </thead> <tbody> <tr><td>80-110MHz</td><td>02</td></tr> <tr><td>110-150MHz</td><td>03</td></tr> <tr><td>150-200MHz</td><td>06</td></tr> <tr><td>200-250MHz</td><td>06</td></tr> <tr><td>250-300MHz</td><td>06</td></tr> <tr><td>300-400MHz</td><td>08</td></tr> <tr><td>400-500MHz</td><td>0b</td></tr> <tr><td>500-600MHz</td><td>0e</td></tr> <tr><td>600-700MHz</td><td>10</td></tr> <tr><td>700-800MHz</td><td>12</td></tr> <tr><td>800-1000MHz</td><td>16</td></tr> <tr><td>1000-1200MHz</td><td>1e</td></tr> <tr><td>1200-1400MHz</td><td>23</td></tr> <tr><td>1400-1600MHz</td><td>2d</td></tr> <tr><td>1600-1800MHz</td><td>32</td></tr> <tr><td>1800-2000MHz</td><td>37</td></tr> <tr><td>2000-2200MHz</td><td>3c</td></tr> <tr><td>2200-2400MHz</td><td>41</td></tr> <tr><td>2400-2500MHz</td><td>46</td></tr> </tbody> </table> | Frequency(1/UI) | Value(HEX) | 80-110MHz | 02 | 110-150MHz | 03 | 150-200MHz | 06 | 200-250MHz | 06 | 250-300MHz | 06 | 300-400MHz | 08 | 400-500MHz | 0b | 500-600MHz | 0e | 600-700MHz | 10 | 700-800MHz | 12 | 800-1000MHz | 16 | 1000-1200MHz | 1e | 1200-1400MHz | 23 | 1400-1600MHz | 2d | 1600-1800MHz | 32 | 1800-2000MHz | 37 | 2000-2200MHz | 3c | 2200-2400MHz | 41 | 2400-2500MHz | 46 |
| Frequency(1/UI) | Value(HEX) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 80-110MHz | 02 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110-150MHz | 03 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 150-200MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 200-250MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 250-300MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 300-400MHz | 08 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 400-500MHz | 0b | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 500-600MHz | 0e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 600-700MHz | 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 700-800MHz | 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 800-1000MHz | 16 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1000-1200MHz | 1e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1200-1400MHz | 23 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1400-1600MHz | 2d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1600-1800MHz | 32 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1800-2000MHz | 37 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2000-2200MHz | 3c | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2200-2400MHz | 41 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2400-2500MHz | 46 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

DPHY RX LANE CK CAL EN

Address: Operational Base + offset (0x0168)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | lane_ck_cal_en Calibration reception enable 1:Enable calibration reception 0:Disable calibration reception |
| 6:0 | RO | 0x0 | Reserved |

DPHY RX LANE 0 MSB

Address: Operational Base + offset (0x01b8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | lane_0_msb MSB enable for pin_rxdatahs_* 1:Enable 0:Disable |
| 5:0 | RO | 0x0 | Reserved |

DPHY RX LANE 0 TTAGO

Address: Operational Base + offset (0x01c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x04 | lane_0_ttago The value of counter for Tta-go of turnaround Tta-go = Ttxclkesc*value |

DPHY RX LANE 0 TTASURE

Address: Operational Base + offset (0x01c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x01 | lane_0_ttasure The value of counter for Tta-sure of turnaround Tta-sure = Ttxclkesc*value |

DPHY RX LANE 0 TTAWAIT

Address: Operational Base + offset (0x01c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x32 | lane_0_ttawait The value of counter for Tta-wait of turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value |

DPHY RX LANE 0 THSSETTLE

Address: Operational Base + offset (0x01e0)

| Bit | Attr | Reset Value | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|------------|-------------|--|-----------------|------------|-----------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|-------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|
| 31:6 | RO | 0x0 | Reserved | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5:0 | RW | 0x1b | <p>lane_0_thssettle The count time of the THS-SETTLE by protocol. after the count done, DPHY will begin to receive the high speed data.</p> <table> <thead> <tr> <th>Frequency(1/UI)</th> <th>Value(HEX)</th> </tr> </thead> <tbody> <tr><td>80-110MHz</td><td>02</td></tr> <tr><td>110-150MHz</td><td>03</td></tr> <tr><td>150-200MHz</td><td>06</td></tr> <tr><td>200-250MHz</td><td>06</td></tr> <tr><td>250-300MHz</td><td>06</td></tr> <tr><td>300-400MHz</td><td>08</td></tr> <tr><td>400-500MHz</td><td>0b</td></tr> <tr><td>500-600MHz</td><td>0e</td></tr> <tr><td>600-700MHz</td><td>10</td></tr> <tr><td>700-800MHz</td><td>12</td></tr> <tr><td>800-1000MHz</td><td>16</td></tr> <tr><td>1000-1200MHz</td><td>1e</td></tr> <tr><td>1200-1400MHz</td><td>23</td></tr> <tr><td>1400-1600MHz</td><td>2d</td></tr> <tr><td>1600-1800MHz</td><td>32</td></tr> <tr><td>1800-2000MHz</td><td>37</td></tr> <tr><td>2000-2200MHz</td><td>3c</td></tr> <tr><td>2200-2400MHz</td><td>41</td></tr> <tr><td>2400-2500MHz</td><td>46</td></tr> </tbody> </table> | Frequency(1/UI) | Value(HEX) | 80-110MHz | 02 | 110-150MHz | 03 | 150-200MHz | 06 | 200-250MHz | 06 | 250-300MHz | 06 | 300-400MHz | 08 | 400-500MHz | 0b | 500-600MHz | 0e | 600-700MHz | 10 | 700-800MHz | 12 | 800-1000MHz | 16 | 1000-1200MHz | 1e | 1200-1400MHz | 23 | 1400-1600MHz | 2d | 1600-1800MHz | 32 | 1800-2000MHz | 37 | 2000-2200MHz | 3c | 2200-2400MHz | 41 | 2400-2500MHz | 46 |
| Frequency(1/UI) | Value(HEX) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 80-110MHz | 02 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110-150MHz | 03 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 150-200MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 200-250MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 250-300MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 300-400MHz | 08 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 400-500MHz | 0b | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 500-600MHz | 0e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 600-700MHz | 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 700-800MHz | 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 800-1000MHz | 16 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1000-1200MHz | 1e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1200-1400MHz | 23 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1400-1600MHz | 2d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1600-1800MHz | 32 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1800-2000MHz | 37 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2000-2200MHz | 3c | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2200-2400MHz | 41 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2400-2500MHz | 46 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

DPHY RX LANE 0 CAL EN

Address: Operational Base + offset (0x01e8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | <p>lane_0_cal_en Calibration reception enable 1:Enable calibration reception 0:Disable calibration reception</p> |
| 6:0 | RO | 0x0 | Reserved |

DPHY RX LANE 1 MSB

Address: Operational Base + offset (0x0238)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--|
| 31:7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | lane_1_msb MSB enable for pin_rxdatahs_* 1:Enable 0:Disable |
| 5:0 | RO | 0x0 | Reserved |

DPHY RX LANE 1 TTAGO

Address: Operational Base + offset (0x0240)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x04 | lane_1_ttago The value of counter for Tta-go of turnaround Tta-go = Ttxclkesc*value |

DPHY RX LANE 1 TTASURE

Address: Operational Base + offset (0x0244)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x01 | lane_1_ttasure The value of counter for Tta-sure of turnaround Tta-sure = Ttxclkesc*value |

DPHY RX LANE 1 TTAWAIT

Address: Operational Base + offset (0x0248)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x32 | lane_1_ttawait The value of counter for Tta-wait of turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value |

DPHY RX LANE 1 THSSETTLE

Address: Operational Base + offset (0x0260)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:6 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|------------|-------------|--|-----------------|------------|-----------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|-------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|
| 5:0 | RW | 0x1b | <p>lane_1_thssettle The count time of the THS-SETTLE by protocol. after the count done, DPHY will begin to receive the high speed data.</p> <table> <thead> <tr> <th>Frequency(1/UI)</th> <th>Value(HEX)</th> </tr> </thead> <tbody> <tr><td>80-110MHz</td><td>02</td></tr> <tr><td>110-150MHz</td><td>03</td></tr> <tr><td>150-200MHz</td><td>06</td></tr> <tr><td>200-250MHz</td><td>06</td></tr> <tr><td>250-300MHz</td><td>06</td></tr> <tr><td>300-400MHz</td><td>08</td></tr> <tr><td>400-500MHz</td><td>0b</td></tr> <tr><td>500-600MHz</td><td>0e</td></tr> <tr><td>600-700MHz</td><td>10</td></tr> <tr><td>700-800MHz</td><td>12</td></tr> <tr><td>800-1000MHz</td><td>16</td></tr> <tr><td>1000-1200MHz</td><td>1e</td></tr> <tr><td>1200-1400MHz</td><td>23</td></tr> <tr><td>1400-1600MHz</td><td>2d</td></tr> <tr><td>1600-1800MHz</td><td>32</td></tr> <tr><td>1800-2000MHz</td><td>37</td></tr> <tr><td>2000-2200MHz</td><td>3c</td></tr> <tr><td>2200-2400MHz</td><td>41</td></tr> <tr><td>2400-2500MHz</td><td>46</td></tr> </tbody> </table> | Frequency(1/UI) | Value(HEX) | 80-110MHz | 02 | 110-150MHz | 03 | 150-200MHz | 06 | 200-250MHz | 06 | 250-300MHz | 06 | 300-400MHz | 08 | 400-500MHz | 0b | 500-600MHz | 0e | 600-700MHz | 10 | 700-800MHz | 12 | 800-1000MHz | 16 | 1000-1200MHz | 1e | 1200-1400MHz | 23 | 1400-1600MHz | 2d | 1600-1800MHz | 32 | 1800-2000MHz | 37 | 2000-2200MHz | 3c | 2200-2400MHz | 41 | 2400-2500MHz | 46 |
| Frequency(1/UI) | Value(HEX) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 80-110MHz | 02 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110-150MHz | 03 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 150-200MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 200-250MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 250-300MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 300-400MHz | 08 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 400-500MHz | 0b | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 500-600MHz | 0e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 600-700MHz | 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 700-800MHz | 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 800-1000MHz | 16 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1000-1200MHz | 1e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1200-1400MHz | 23 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1400-1600MHz | 2d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1600-1800MHz | 32 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1800-2000MHz | 37 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2000-2200MHz | 3c | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2200-2400MHz | 41 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2400-2500MHz | 46 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

DPHY RX LANE 1 CAL EN

Address: Operational Base + offset (0x0268)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | <p>lane_1_cal_en Calibration reception enable 1:Enable calibration reception 0:Disable calibration reception</p> |
| 6:0 | RO | 0x0 | Reserved |

DPHY RX LANE 2 MSB

Address: Operational Base + offset (0x02b8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | <p>lane_2_msb MSB enable for pin_rxdatahs_* 1:Enable 0:Disable</p> |
| 5:0 | RO | 0x0 | Reserved |

DPHY RX LANE 2 TTAGO

Address: Operational Base + offset (0x02c0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x04 | lane_2_ttago The value of counter for Tta-go of turnaround Tta-go = Ttxclkesc*value |

DPHY RX LANE 2 TTASURE

Address: Operational Base + offset (0x02c4)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x01 | lane_2_ttasure The value of counter for Tta-sure of turnaround Tta-sure = Ttxclkesc*value |

DPHY RX LANE 2 TTAWAIT

Address: Operational Base + offset (0x02c8)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x32 | lane_2_ttawait The value of counter for Tta-wait of turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value |

DPHY RX LANE 2 THSSETTLE

Address: Operational Base + offset (0x02e0)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:6 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|------------|-------------|--|-----------------|------------|-----------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|-------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|
| 5:0 | RW | 0x1b | <p>lane_2_thssettle The count time of the THS-SETTLE by protocol. after the count done, DPHY will begin to receive the high speed data.</p> <table> <thead> <tr> <th>Frequency(1/UI)</th> <th>Value(HEX)</th> </tr> </thead> <tbody> <tr><td>80-110MHz</td><td>02</td></tr> <tr><td>110-150MHz</td><td>03</td></tr> <tr><td>150-200MHz</td><td>06</td></tr> <tr><td>200-250MHz</td><td>06</td></tr> <tr><td>250-300MHz</td><td>06</td></tr> <tr><td>300-400MHz</td><td>08</td></tr> <tr><td>400-500MHz</td><td>0b</td></tr> <tr><td>500-600MHz</td><td>0e</td></tr> <tr><td>600-700MHz</td><td>10</td></tr> <tr><td>700-800MHz</td><td>12</td></tr> <tr><td>800-1000MHz</td><td>16</td></tr> <tr><td>1000-1200MHz</td><td>1e</td></tr> <tr><td>1200-1400MHz</td><td>23</td></tr> <tr><td>1400-1600MHz</td><td>2d</td></tr> <tr><td>1600-1800MHz</td><td>32</td></tr> <tr><td>1800-2000MHz</td><td>37</td></tr> <tr><td>2000-2200MHz</td><td>3c</td></tr> <tr><td>2200-2400MHz</td><td>41</td></tr> <tr><td>2400-2500MHz</td><td>46</td></tr> </tbody> </table> | Frequency(1/UI) | Value(HEX) | 80-110MHz | 02 | 110-150MHz | 03 | 150-200MHz | 06 | 200-250MHz | 06 | 250-300MHz | 06 | 300-400MHz | 08 | 400-500MHz | 0b | 500-600MHz | 0e | 600-700MHz | 10 | 700-800MHz | 12 | 800-1000MHz | 16 | 1000-1200MHz | 1e | 1200-1400MHz | 23 | 1400-1600MHz | 2d | 1600-1800MHz | 32 | 1800-2000MHz | 37 | 2000-2200MHz | 3c | 2200-2400MHz | 41 | 2400-2500MHz | 46 |
| Frequency(1/UI) | Value(HEX) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 80-110MHz | 02 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110-150MHz | 03 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 150-200MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 200-250MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 250-300MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 300-400MHz | 08 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 400-500MHz | 0b | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 500-600MHz | 0e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 600-700MHz | 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 700-800MHz | 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 800-1000MHz | 16 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1000-1200MHz | 1e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1200-1400MHz | 23 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1400-1600MHz | 2d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1600-1800MHz | 32 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1800-2000MHz | 37 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2000-2200MHz | 3c | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2200-2400MHz | 41 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2400-2500MHz | 46 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

DPHY RX LANE 2 CAL EN

Address: Operational Base + offset (0x02e8)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | <p>lane_2_cal_en Calibration reception enable 1:Enable calibration reception 0:Disable calibration reception</p> |
| 6:0 | RO | 0x0 | Reserved |

DPHY RX LANE 3 MSB

Address: Operational Base + offset (0x0338)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|---|
| 31:7 | RO | 0x0 | Reserved |
| 6 | RW | 0x0 | <p>lane_3_msb MSB enable for pin_rxdatahs_* 1:Enable 0:Disable</p> |
| 5:0 | RO | 0x0 | Reserved |

DPHY RX LANE 3 TTAGO

Address: Operational Base + offset (0x0340)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x04 | lane_3_ttago The value of counter for Tta-go of turnaround Tta-go = Ttxclkesc*value |

DPHY RX LANE 3 TTASURE

Address: Operational Base + offset (0x0344)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x01 | lane_3_ttasure The value of counter for Tta-sure of turnaround Tta-sure = Ttxclkesc*value |

DPHY RX LANE 3 TTAWAIT

Address: Operational Base + offset (0x0348)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|---|
| 31:6 | RO | 0x0 | Reserved |
| 5:0 | RW | 0x32 | lane_0_ttawait The value of counter for Tta-wait of turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value |

DPHY RX LANE 3 THSSETTLE

Address: Operational Base + offset (0x0360)

| Bit | Attr | Reset Value | Description |
|------------|-------------|--------------------|--------------------|
| 31:6 | RO | 0x0 | Reserved |

| Bit | Attr | Reset Value | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|------------|-------------|--|-----------------|------------|-----------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|------------|----|-------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|--------------|----|
| 5:0 | RW | 0x1b | <p>lane_3_thssettle The count time of the THS-SETTLE by protocol. after the count done, DPHY will begin to receive the high speed data.</p> <table> <thead> <tr> <th>Frequency(1/UI)</th> <th>Value(HEX)</th> </tr> </thead> <tbody> <tr><td>80-110MHz</td><td>02</td></tr> <tr><td>110-150MHz</td><td>03</td></tr> <tr><td>150-200MHz</td><td>06</td></tr> <tr><td>200-250MHz</td><td>06</td></tr> <tr><td>250-300MHz</td><td>06</td></tr> <tr><td>300-400MHz</td><td>08</td></tr> <tr><td>400-500MHz</td><td>0b</td></tr> <tr><td>500-600MHz</td><td>0e</td></tr> <tr><td>600-700MHz</td><td>10</td></tr> <tr><td>700-800MHz</td><td>12</td></tr> <tr><td>800-1000MHz</td><td>16</td></tr> <tr><td>1000-1200MHz</td><td>1e</td></tr> <tr><td>1200-1400MHz</td><td>23</td></tr> <tr><td>1400-1600MHz</td><td>2d</td></tr> <tr><td>1600-1800MHz</td><td>32</td></tr> <tr><td>1800-2000MHz</td><td>37</td></tr> <tr><td>2000-2200MHz</td><td>3c</td></tr> <tr><td>2200-2400MHz</td><td>41</td></tr> <tr><td>2400-2500MHz</td><td>46</td></tr> </tbody> </table> | Frequency(1/UI) | Value(HEX) | 80-110MHz | 02 | 110-150MHz | 03 | 150-200MHz | 06 | 200-250MHz | 06 | 250-300MHz | 06 | 300-400MHz | 08 | 400-500MHz | 0b | 500-600MHz | 0e | 600-700MHz | 10 | 700-800MHz | 12 | 800-1000MHz | 16 | 1000-1200MHz | 1e | 1200-1400MHz | 23 | 1400-1600MHz | 2d | 1600-1800MHz | 32 | 1800-2000MHz | 37 | 2000-2200MHz | 3c | 2200-2400MHz | 41 | 2400-2500MHz | 46 |
| Frequency(1/UI) | Value(HEX) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 80-110MHz | 02 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110-150MHz | 03 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 150-200MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 200-250MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 250-300MHz | 06 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 300-400MHz | 08 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 400-500MHz | 0b | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 500-600MHz | 0e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 600-700MHz | 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 700-800MHz | 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 800-1000MHz | 16 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1000-1200MHz | 1e | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1200-1400MHz | 23 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1400-1600MHz | 2d | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1600-1800MHz | 32 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1800-2000MHz | 37 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2000-2200MHz | 3c | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2200-2400MHz | 41 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2400-2500MHz | 46 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

DPHY RX LANE 3 CAL EN

Address: Operational Base + offset (0x0368)

| Bit | Attr | Reset Value | Description |
|------|------|-------------|--|
| 31:8 | RO | 0x0 | Reserved |
| 7 | RW | 0x0 | <p>lane_3_cal_en Calibration reception enable 1:Enable calibration reception 0:Disable calibration reception</p> |
| 6:0 | RO | 0x0 | Reserved |

16.4 Interface Description

Table 16-1 MIPI DPHY RX Interface Description

| Module Pin | Direction | Pad Name | IOMUX Setting |
|------------|-----------|--------------------|---------------|
| clkp | I | IO_DPHY_RX0_CLKP | |
| clkn | I | IO_DPHY_RX0_CLKN | |
| datap0 | I | IO_DPHY_RX0_DATAP0 | |
| datan0 | I | IO_DPHY_RX0_DATAN0 | |
| datap1 | I | IO_DPHY_RX0_DATAP1 | |
| datan1 | I | IO_DPHY_RX0_DATAN1 | |
| datap2 | I | IO_DPHY_RX0_DATAP2 | |
| datan2 | I | IO_DPHY_RX0_DATAN2 | |

| Module Pin | Direction | Pad Name | IOMUX Setting |
|------------|-----------|--------------------|---------------|
| datap3 | I | IO_DPHY_RX0_DATAP3 | |
| datan3 | I | IO_DPHY_RX0_DATAN3 | |

16.5 Application Notes

Some important signals are from GRF, for example: csiphy_forcerxmode_0/1/2/3 = bus_grf_vi_con0[3:0], csiphy_datalane_en_0/1/2/3 = bus_grf_vi_con0[7:4], csiphy_clklane_en = bus_grf_vi_con0[8]. And some important status signals could drive the GRF, for example: bus_grf_vi_status[3:0]=csiphy_ulpssactivenot_0/1/2/3.

The following is a step by step instruction for MIPI DPHY RX in normal working mode

Setup Steps:

Step1: Send 0x7d to register 0x00. Enable the DPHY

Step2: Initialization completed.

Rockchip Confidential